Transactions of the I·R·E

Professional Group on ELECTRONIC COMPUTERS

VOLUME EC-3 DECEMBER 1954 NUMBER 4

Published Quarterly

e Institute of Radio Engineers

Transactions of the I•R•E®

Professional Group on Electronic Computers

Published by the Institute of Radio Engineers, Inc., for the Professional Group on Electronic Computers at 1 East 79th Street, New York 21, N.Y. Responsibility for the contents rests upon the authors and not upon the Institute, the Group, or its Members. Extra copies of this issue are available for sale to IRE-PGEC members at $1.10; to other IRE members at $1.65; and to nonmembers at $3.30. Address requests to The Institute of Radio Engineers, 1 East 79th Street, New York 21, N.Y.

Notice to Authors: Address all papers and editorial correspondence to R. E. Meagher, 168 Engineering Research Laboratory, University of Illinois, Urbana, Ill. To avoid delay, 3 copies of papers and figures should be submitted, together with the originals of the figures which will be returned on request. All material will be returned if a paper is not accepted.

The Use of a Reflected Code in Digital Control Systems

F. A. FOSS*

Summary—The reflected (Gray) binary code has previously been used primarily in analog-to-digital conversion applications. However, the reflected binary code has, in addition, characteristics which strongly recommend it as a design factor in the synthesis of digital elements within closed-loop control systems. This paper describes several designs of typical reflected binary switching circuits. These switching circuits are, specifically, a reflected-to-conventional binary translator, reflected binary-to-analog converters, and a reflected binary contactor comparator. All of the circuit designs are logically related to the form of the reflected-to-conventional binary translator. These switching circuits can be used as digital control elements in the described original types of digital positional servomechanisms and digital accumulators.

Characteristics of the Reflected Binary Code

F. GRAY of the Bell Telephone Laboratories conceived the use of a reflected binary number system.[1] He designed a reflected binary coding mask for the cathode ray tube portion of a pulse code communication system. He was interested in converting analog deflection signals into reflected binary code groups. The reflected binary code has been used in many other analog-to-digital conversion devices requiring a dynamic read out of the converted analog quantity. This paper will show that the reflected binary code also has useful characteristics applicable to the design of switching circuits used in unique digital servomechanisms. The logic used in the synthesis of these circuits is better understood if we first review the defining characteristics of the reflected binary number system.

The reflected binary code has the important property that only one code position changes from 0 to 1, or vice versa, when the decimal equivalent value changes by one unit. The reflected binary code can be derived from the conventional binary, as indicated in Table I. The algebraic postulates define half addition which is used to derive this code from the conventional binary code. The translation of conventional binary-to-reflected binary, by means of half addition of the appropriate reflected binary coefficients, is also defined in Table I. Note that a binary digit, 1, is present in any conventional binary code position if there are an odd number of 1 digits to be found, inclusive of the same and all higher code positions in the equivalent reflected binary code. Note also, that a binary digit, 0, is present in any conventional binary code position if there are an even number of 1 digits to be found, inclusive of the same and all higher code positions in the equivalent reflected binary code. These properties will be used in later designs. Table II is illustrative of typical equivalent conventional and reflected binary codes.

* Project Engineer, IBM Corp., Endicott, N. Y.

[1] F. Gray, "Pulse Code Communication," Patent 2,632,058; March 17, 1953.

TABLE I

Reflected and Conventional Binary Code Translation Formulas

C = Conventional Binary Number
R = Reflected Binary Number
C_p = Conventional Binary Term in Position p
R_p = Reflected Binary Term in Position p
$C = C_nC_{n-1}C_{n-2}\cdots C_3C_2C_1C_0$
$R = R_nR_{n-1}R_{n-2}\cdots R_3R_2R_1R_0$

Reflected-To-Conventional Binary Translation

$$C_n = R_n$$
$$C_{n-1} = R_n + R_{n-1}$$
$$C_{n-2} = R_n + R_{n-1} + R_{n-2}$$
$$\vdots$$
$$C_2 = R_n + R_{n-1} + R_{n-2} + R_{n-3} + \cdots + R_4 + R_3 + R_2$$
$$C_1 = R_n + R_{n-1} + R_{n-2} + R_{n-3} + \cdots + R_4 + R_3 + R_2 + R_1$$
$$C_0 = R_n + R_{n-1} + R_{n-2} + R_{n-3} + \cdots + R_4 + R_3 + R_2 + R_1 + R_0$$

general formula $$C_s = \sum_{t=s}^{n} R_t$$

Conventional-To-Reflected Binary Translation

$$R_n = C_n$$
$$R_{n-1} = C_n + C_{n-1}$$
$$R_{n-2} = C_{n-1} + C_{n-2}$$
$$\vdots$$
$$R_2 = C_3 + C_2$$
$$R_1 = C_2 + C_1$$
$$R_0 = C_1 + C_0$$

general formula $$R_s = C_{s+1} + C_s$$

Algebraic Postulates

$0+0=0$
$0+1=1$
$1+0=1$
$1+1=0$

TABLE II

Equivalent Conventional and Reflected Binary Codes

Decimal Number	Conventional Binary Code Positions	Reflected Binary Code Positions
	4 3 2 1 0	4 3 2 1 0
0	0 0 0 0 0	0 0 0 0 0
1	0 0 0 0 1	0 0 0 0 1
2	0 0 0 1 0	0 0 0 1 1
3	0 0 0 1 1	0 0 0 1 0
4	0 0 1 0 0	0 0 1 1 0
5	0 0 1 0 1	0 0 1 1 1
6	0 0 1 1 0	0 0 1 0 1
7	0 0 1 1 1	0 0 1 0 0
8	0 1 0 0 0	0 1 1 0 0
9	0 1 0 0 1	0 1 1 0 1
10	0 1 0 1 0	0 1 1 1 1
11	0 1 0 1 1	0 1 1 1 0
12	0 1 1 0 0	0 1 0 1 0
13	0 1 1 0 1	0 1 0 1 1
14	0 1 1 1 0	0 1 0 0 1
15	0 1 1 1 1	0 1 0 0 0
16	1 0 0 0 0	1 1 0 0 0

Since only a single code position changes from 1 to 0, or vice versa, when a reflected binary coded number changes in magnitude by 1, an ambiguity in the changing code position is of minor importance. A change in a higher order reflected binary code position has no more numerical value significance than a change in a lower order code position. One, or all, code positions may change from 1 to 0, or vice versa, when a conventional binary coded number changes in magnitude by 1. A conventional binary code position has the numerical value, 2^n, where n indicates the corresponding code position. An ambiguity in any of the changing conventional binary code positions, therefore, may be of major importance. Reflected binary coded mechanisms are more adaptable, therefore, to dynamic parallel readout systems. The need for detent mechanisms is eliminated.

Reflected-to-Conventional Binary Translator

The designs of all of the reflected binary control elements described in this paper are logically related to the form of the switching circuit which translates the reflected to the conventional binary code. The relay reflected-to-conventional binary translator is a symmetric network.[2] A symmetric network may be defined as one in which the requirements for closure may be given in terms of the number of relays operated.[3] Since the translation property of the reflected code involves even and odd numbers of 1's within all considered code positions rather than specific code positions, a symmetric type translating circuit is required.

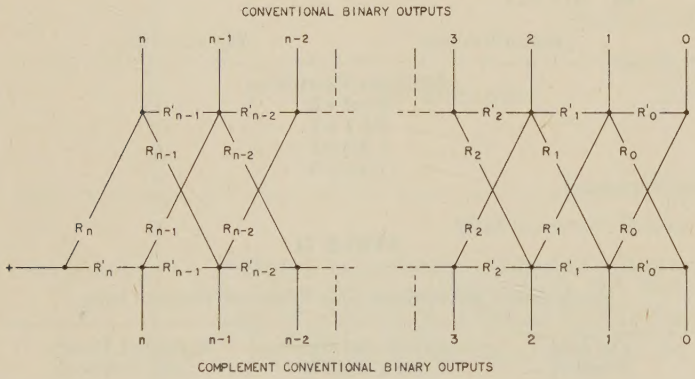


Fig. 1—Logical development of reflected-to-conventional binary translator.

The logical development of the multi-terminal, reflected-to-conventional binary translator is indicated in Fig. 1. The translator has the same structure for an arbitrary number of code positions. The subscripts in Fig. 1 are used to denote code position. The primes are used to denote normally closed-relay contact points. The unprimed quantities denote normally open-relay contact points. The upper set of terminals yields the translated-reflected binary in conventional binary form. The lower set of terminals yields the 1's complements of the conventional binary number. The electrical details of the reflected-to-conventional binary relay translator are given in Fig. 2. Reflected-to-conventional binary translators have also been constructed using diodes and transistors.[4]

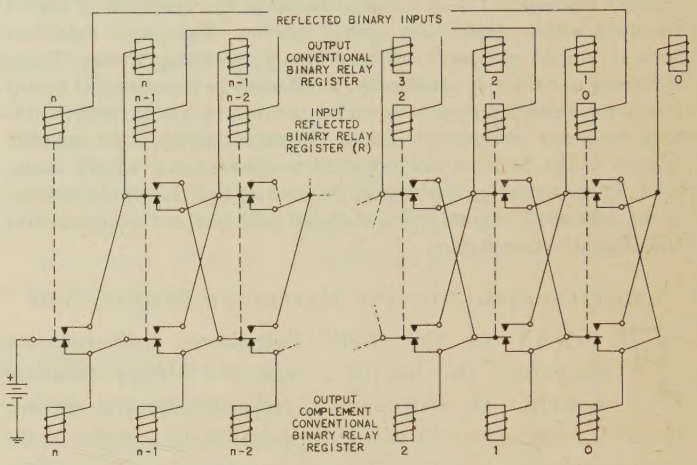


Fig. 2—Reflected-to-conventional binary relay translator.

Reflected Binary-to-Analog Converters

A useful digital control element is a simple reflected binary-to-analog converter constructed solely of relay contact points and resistors. The logical development of the reflected binary-to-analog converter is given in Fig. 3. This design results from a consideration of the

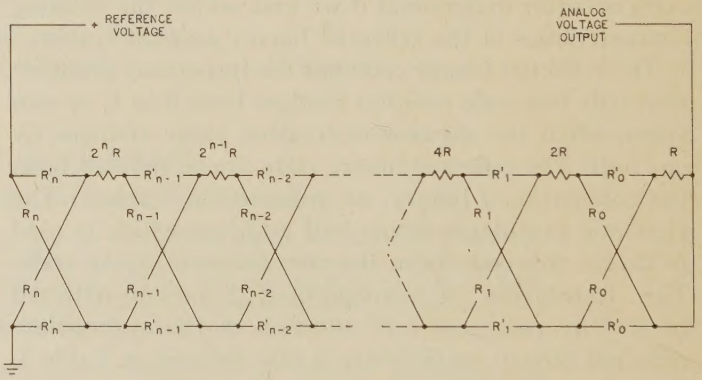


Fig. 3—Logical development of reflected binary-to-analog converter.

reflected-to-conventional binary translator. One set of resistors is used. The impedance of any resistor is chosen as 2^nR, where n is the corresponding code position and R is the value of the resistor in the 0 code position. The basic network is a symmetric network similar to the previously described translator. The electrical details of the reflected binary-to-analog converter are given in Fig. 4 (facing page).

[2] B. Lippel, "Interconnections of analog and digital data in systems for measurement and control." *Proc. NEC*, vol. 8, pp. 636–646; Sept. 29–Oct. 1, 1952.

[3] W. Keister, A. E. Ritchie, and S. H. Washburn, "The Design of Switching Circuits," D. Van Nostrand, Company, Inc., New York, N. Y.; 1951.

[4] R. E. Yaeger, "The Gray-to-binary translator and shift register," *The Transistor*, Bell Telephone Labs., pp. 611–626; November, 1951.

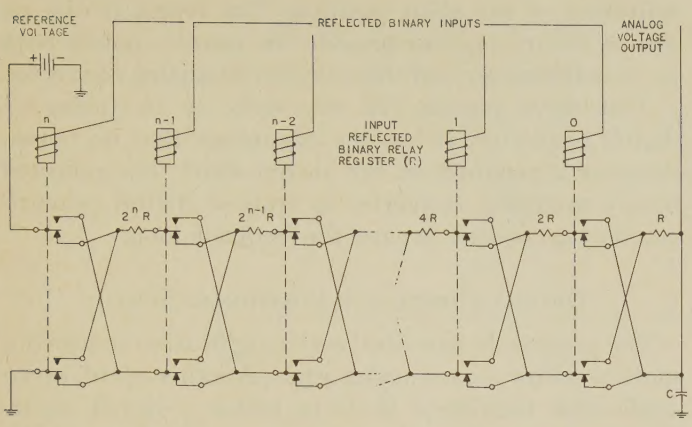
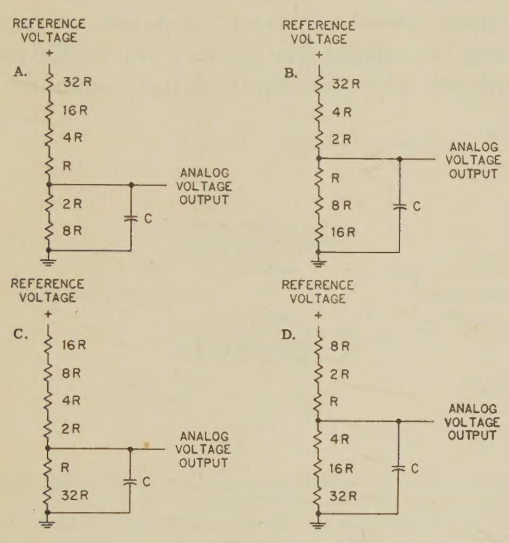


Fig. 4—Reflected binary-to-analog relay converter.

The equivalent circuit of the reflected binary-to-analog converter for several input reflected binary codes is given in Fig. 5. Note that a constant impedance is presented to the reference voltage supply by the converter. This network switches those resistors representing the presence of binary 1's in the conventional binary (translated-reflected) code into a series connection between the analog voltage output terminal and ground.



Example	Reflected Binary Code Positions	Equivalent Conventional Binary Code Positions	Equivalent Decimal Number
	5 4 3 2 1 0	5 4 3 2 1 0	
A	0 0 1 1 1 1	0 0 1 0 1 0	10
B	0 1 0 1 0 1	0 1 1 0 0 1	25
C	1 1 0 0 0 1	1 0 0 0 0 1	33
D	1 0 1 1 1 0	1 1 0 1 0 0	52

Equivalent Circuit of Reflected Binary-to-Analog Relay Converter for Given Input Reflected Binary Codes

Fig. 5—Circuit analysis of six-position reflected binary-to-analog relay converter.

This network also switches those resistors representing the presence of binary 0's in the conventional binary (translated-reflected) code into a series connection between the analog voltage output terminal and the reference supply voltage.

Another similar form of the reflected binary-to-analog converter can be designed, which differs from the one described above, in that it would present a constant impedance to the load. A single set of resistors would again be used. The logical development of this reflected binary-to-analog converter is given in Fig. 6. Note that the impedance of the resistor in the lowest order code position is R. The impedance of a resistor in the nth order code position is $R/2^n$.

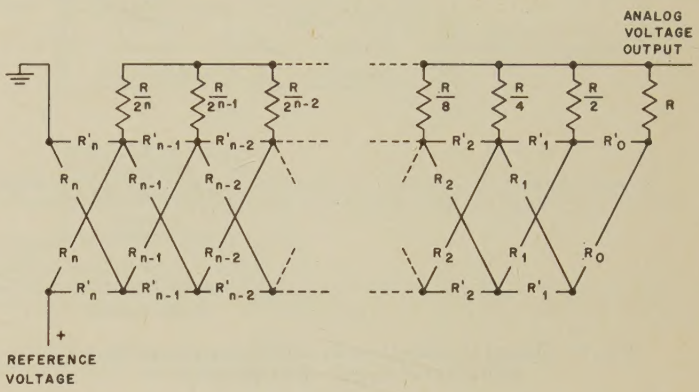


Fig. 6—Logical development of alternate-reflected binary-to-analog converter.

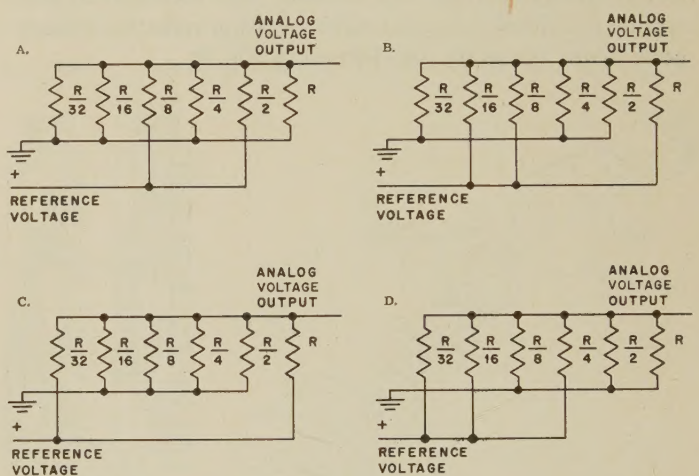


Fig. 7—Equivalent circuit of reflected binary-to-analog relay converter for given input reflected binary codes.

The equivalent circuits of this alternate-reflected binary-to-analog converter for several input-reflected binary codes are given in Fig. 7. The input-reflected binary codes given in Fig. 5 are used in Fig. 7. Note that a constant impedance is presented to the load circuit by the converter. This network switches those resistors representing the presence of binary 1's in the conventional binary (translated-reflected) code into a parallel connection between the analog voltage output terminal and the reference supply voltage. This network switches those resistors, representing the presence of binary 0's in the conventional binary (translated-reflected) code, into a parallel connection between the analog voltage output terminal and ground.

Digital Positional Servo

The reflected binary-to-analog converters, as previously described, can be used in the feedback loop of an original digital positional servo, as indicated in Fig. 8.

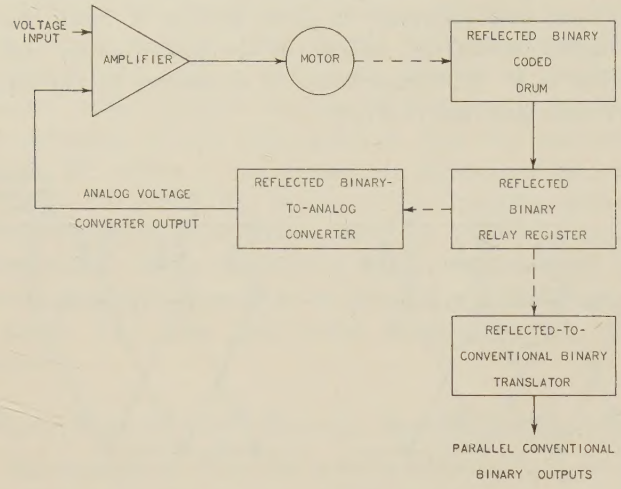


Fig. 8—Digital positional-servo with conventional binary indication of output shaft position.

The output shaft position is converted into a reflected code representation by means of code drums, or any other suitable coding mechanism. The mechanical and coding details of a typical eight-position reflected binary code drum assembly are defined in Fig. 9.

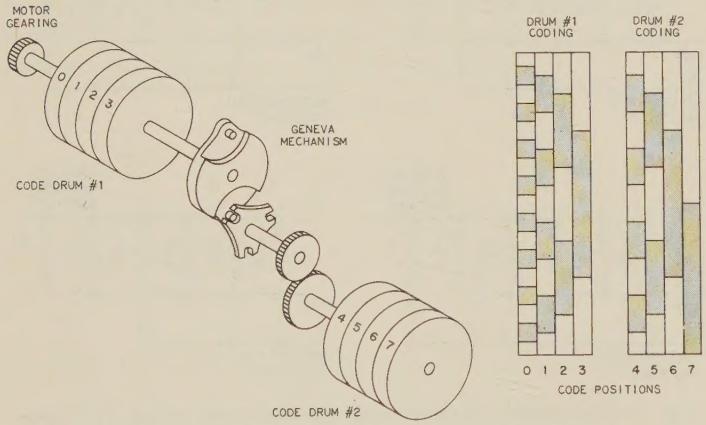


Fig. 9—Reflected binary code drums.

A pattern of 32 codes is machined on the four-position code drum number 1. A pattern of 16 codes is employed on the four-position code drum number 2. A 120-degree motion of the geneva output shaft occurs twice per one revolution of the geneva input shaft. A gearing ratio of 3:16 is used from the geneva output shaft to the second drum. An advantage of this type of coding mechanism design is the resultant small diameter of the individual code drums. A mechanical stop should also be provided to prevent the shaft of code drum number 2 from rotating through its 360-degree position. Single-make contacts, actuated by reading fingers riding on the reflected binary code drums, control the operation of the relays in the reflected binary relay register. The reflected-to-conventional binary translator can also be used, as indicated in Fig. 8, to give a digital output representation of the shaft position. The relays in the reflected binary register provide the contact points both for this translator and the reflected-to-analog converter.

This servo system can also serve as an analog-to-digital converter. It has the advantage that no potentiometer is required on the output shaft. The reflected binary-to-analog converter is truly a digital potentiometer that is used to close the feedback loop.

Digital Contactor Positional Servo

The previously described servo application required a shaft to follow a continuous input. Another useful servo application requires a shaft to follow a digital input. This control problem leads to a very novel use of the reflected binary code.

A digital contactor positional servo can be designed, as illustrated in Fig. 10, using an eight-position contactor-type comparator. The function of the comparator is simply to determine whether the code, as read from the output shaft-coding mechanism, is higher or lower in magnitude than the input code. A contactor-type comparator is one in which discontinuous (on-off), rather than continuous, control signals are generated. The digital contactor positional servo, as shown, is capable of positioning the output shaft to any one of 256 positions in accordance with the input digital command signal.

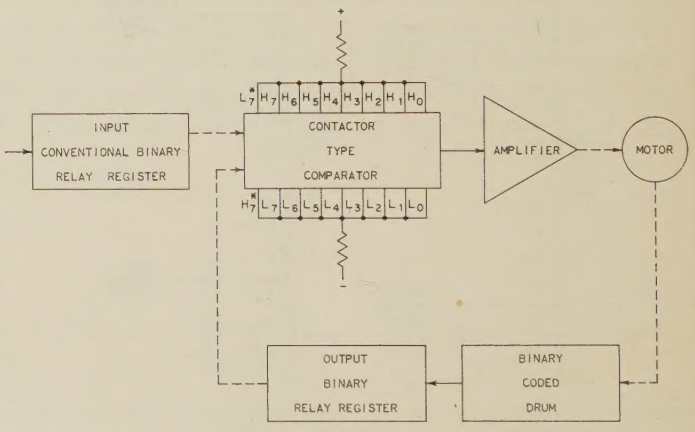
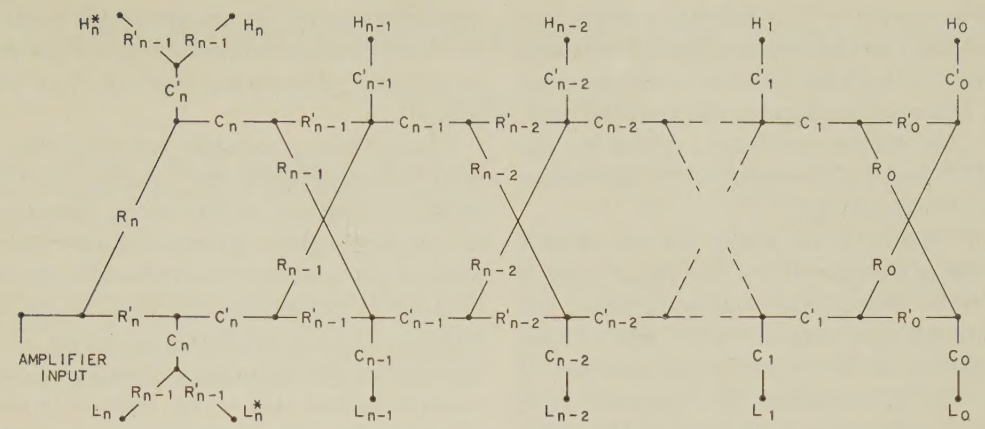


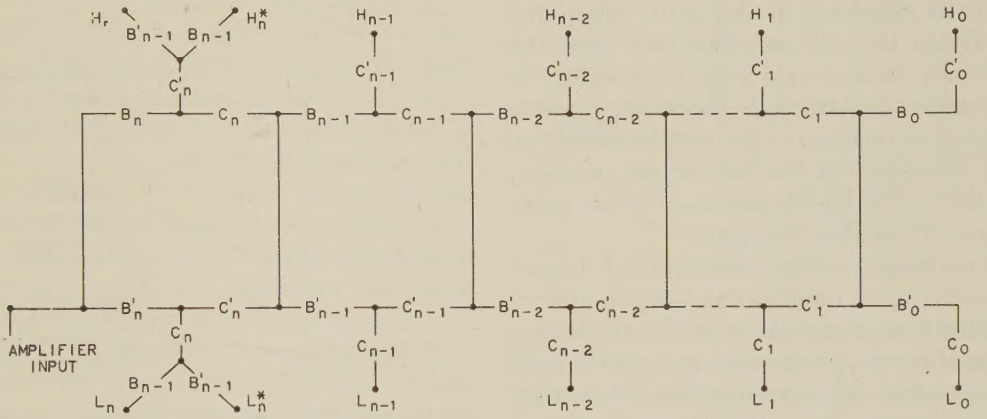
Fig. 10—Digital contactor positional-servo.

The logical developments of two general forms of the comparator are shown (Figs. 11, 12, facing page). These circuits are similar in form to the reflected-to-conventional binary translator. The functioning of one comparator is based on the use of a reflected binary coded drum on the output shaft (Fig. 11). The other comparator requires a conventional binary-coded drum on the output shaft (Fig. 12). Either comparator furnishes a set of terminals (*H*, high and *L*, low), which can, by proper connection, provide varied types of contactor control. In general, the connections, as shown in Fig. 10, provide for positive or negative drive of the motor, if the output shaft position is respectively higher or lower than the corresponding input angular position. However, an exception to this type of control, which is necessary for proper stabilization, is noted in the next paragraph.



C_s = s^{th} code position of input conventional binary-coded number.
R_s = s^{th} code position of output reflected binary-coded number.
H_r = Indication that | output angular shaft position | > | input angular shaft position |. Presence of r^{th} conventional binary code position in the translated output reflected binary code. Absence of r^{th} conventional binary code position in the input conventional binary code and all higher-order code positions are equal ($r \neq n$).
H_n = Same as above, except that output shaft is in zone 3 and input shaft is in zone 1 or 2 ($r \neq n$).
H_n^* = Same as H_r, except that output shaft is in zone 4 and input shaft is in zone 1 or 2 ($r = n$).
L_r = Indication that | output angular shaft position | < | input angular shaft position |. Absence of r^{th} conventional binary code position in the translated output reflected binary code. Presence of r^{th} conventional binary code position in the input conventional binary code and all higher-order code positions are equal ($r \neq n$).
L_n = Same as above, except that output shaft is in zone 2 and input shaft is in zone 3 or 4 ($r = n$).
L_n^* = Same as above, except that output shaft is in zone 1 and input shaft is in zone 3 or 4 ($r = n$).

Fig. 11—Logical development of contactor-type comparator.



C_s = s^{th} code position of input conventional binary-coded number.
B_s = s^{th} code position of output conventional binary-coded number.
H_r = Indication that | output angular shaft position | > | input angular shaft position |. Presence of r^{th} conventional binary code position in the output conventional binary code. Absence of r^{th} conventional binary code position in the input conventional binary code and all higher code positions are equal ($r \neq n$).
H_n = Same as above, except that output shaft is in zone 3 and input shaft is in zone 1 or 2 ($r = n$).
H_n^* = Same as H_r, except that output shaft is in zone 4 and input shaft is in zone 1 or 2 ($r = n$).
L_r = Indication that | output angular shaft position | < | input angular shaft position |. Absence of r^{th} conventional binary code position in the output conventional binary code. Presence of r^{th} conventional binary code position in the input conventional binary code and all higher code positions are equal ($r \neq n$).
L_n = Same as above, except that output shaft is in zone 2 and input shaft is in zone 3 or 4 ($r = n$).
L_n^* = Same as above, except that output shaft is in zone 1 and input shaft is in zone 3 or 4 ($r = n$).

Fig. 12—Logical development of alternate contactor-type comparator.

A special stability problem is encountered near or at the 360-degree position of the output shaft. Here, for example, in an eight-position binary device, the reflected code changes abruptly from 00000000 to 10000000. Consider the case where the input binary code is 00000000. If the motor, in positioning binary-coded drums, drives the output shaft past the desired 00000000 position in the direction of decreasing coded magnitudes, a comparator constructed only on a high-low numerical value basis, would, for control purposes, always have a "high" reading. The motor would drive continuously and never position the output shaft to the static desired position. This unstable condition requires creating a region about the 00000000–10000000 reflected code positions or the 00000000–11111111 conventional code positions, so that proper control action can be obtained. A stable system can be achieved rather simply. The binary code positions defining the 360-degree travel of the output shaft are grouped into 4 zones. Considering the highest two conventional code positions, the 00 code defines zone 1, the 01 code defines zone 2, the 10 code defines zone 3, and the 11 code defines zone 4. Considering the highest two reflected code positions, the 00 code defines zone 1, the 01 code defines zone 2, the 11 code defines zone 3,

and the 10 code defines zone 4. The high-low comparator outputs, corresponding to the highest code positions, can now be modified by this zone information to produce a stable system. The switching logic of the zone concept, that provides for this stable positional system by the use of terminals H_n* and L_n*, is included in the design illustrated in Figs. 10, 11 and 12.

The comparator circuits sense whenever the output shaft coded position is in zone 4 and the input code is in zone 1 or 2. Under these positional conditions, the comparator circuits energize terminal H_n*. The output shaft position relative to the input code position is "high"; however, the H_n* terminal is connected as if the codes were "low" relative to each other. (This connection provides for the reversing of the motor when the output shaft is driven through an input code near the 0-degree position in zone 1 into zone 4.)

The comparator circuits also sense whenever the output shaft coded position is in zone 1 and the input code is in zone 3 or 4. Under these positional conditions, the comparator circuits energize terminal L_n*. The output shaft position relative to the input code position is now "low"; however, the L_n* terminal is connected as if the codes were "high" relative to each other. (This connection provides for the reversing of the motor when the output shaft is driven through an input code near the 360-degree position in zone 4 into zone 1.) Stability is achieved, and positional control is therefore possible through the 360-degree position of the output shaft. No mechanical stop is needed at the 360-degree position, contrary to the need of the digital positional servo using the reflected binary-to-analog converter.

The comparator design, using conventional binary codes for input and output, requires the least number of contact points. However, a conventional binary design leads to a positional servo system that is unstable even with careful mechanical adjustment. This "hunting" phenomena is a result of the inability to adjust, mechanically, all code drum contacts to make and break simultaneously. Conventional binary code drums, in changing from the 00010000 code to the lower valued 00001111 code, might, for example, have an erroneous intermediate higher valued 00011000 code if the make-contact for the third code position closes before any of the other code positions change. Such false intermediate conventional binary codes can cause the servo system to oscillate about in incorrect shaft positions. The use of a reflected binary code mechanism and comparator removes this control difficulty. False intermediate codes cannot occur. The digital contactor positional servo, with a reflected binary contactor type comparator, has stable operational characteristics.

Digital Accumulator

Another application of the reflected-to-conventional binary translator and reflected binary code drums is indicated in Fig. 13. This device would be capable of adding to the sum, in conventional binary form in the accumulator, quantities in reflected binary form which are functions of the angular shaft position. The control of this digital accumulator is not shown. The addition would take place under control of an external signaling source.

Three binary registers are required: one register to hold the reflected binary number as received from the drum, its contact points to be used for the translator; the second register to serve as an input register for the parallel conventional binary adder, its contact points to be used to synthesize the logic of the adder; the third register, the accumulator, to serve as the other input register for the parallel conventional binary adder, its contact points also to be used to synthesize the logic of the adder. The first register serves a dual purpose. It also temporarily stores the result of the addition so that the accumulator register can be cleared before receiving the new sum.

Note that with parallel transfer of reflected binary information from the drum, detent action is no longer necessary. The drum contacts would be isolated from the input register during the arithmetic process.

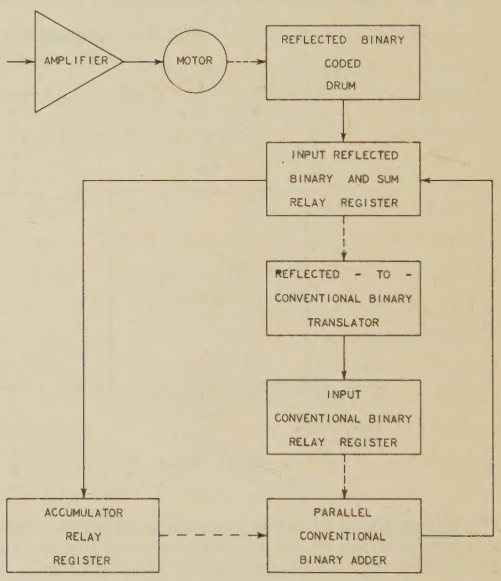


Fig. 13—Digital accumulator.

Conclusions

The reflected code is readily converted to its equivalent conventional binary code by the previously discussed translator. The reflected code is easily converted into its equivalent analog voltage as previously noted. The reflected binary-to-analog converter can be designed so that it either presents a constant impedance to the reference voltage source or to the load circuit. The reflected binary-to-analog converters can be used to synthesize digital positional servos which can also be used as analog-to-digital converters. A stable positional-servo can be designed, having a digital input and a reflected binary comparator. The circuit simplicity of the described translator, converters and comparators, and the desirable operational characteristics of the reflected binary code can be combined into several control applications heretofore solved by more complex means.

A Transistorized Pulse Code Modulator

G. R. PARTRIDGE*

Summary—A pulse code modulator is described in which transistors are the only active circuit components. The functions of quantizing and encoding the signal are performed entirely by semiconductor diodes, using transistor amplifiers where necessary to increase the signal level. The readout is obtained from a set of "and" gates, each gate having as one of its inputs an output from the encoding system, and as the other input, a pulse obtained from a time delay circuit to assure sequential operation. The system produces a three digit binary code at a sampling rate of 5,000 cps.

Introduction

PULSE code modulation (PCM) is inherently a system of analog to digital conversion. A continuously variable function, such as a voltage, is impressed upon the input of the modulator, and some definite pattern of pulses appears at the output in response to the magnitude of the input signal. Since, with a finite number of pulses permitted in the output, only a finite number of levels can be distinguished in the input, it is apparent that one requirement of a PCM system is the "quantizing" of the input, *i.e.*, separating it into discrete levels that can correspond directly with a given pulse pattern. Having done this, the circuit must then select the appropriate pulse sequence from a set of ready-made choices, as in the Sears encoding tube, or proceed to generate one from scratch, as it were, by any of a number of methods.[1] This step is known as "encoding." Finally, the pulses must be "read out," *i.e.*, delivered to the output terminals. This might be done as part of the encoding process, as in the Sears tube, or as a separate operation controlled by a series of gates.

This paper describes a method for carrying out these three steps by circuits using only transistors, crystal diodes, and ordinary passive resistors and capacitors. The quantizing and encoding are done simultaneously—a "parallel" operation in computer terminology—while the readout must be sequential or "serial" in nature.

The Quantizing System

Fig. 1 illustrates the quantizing circuit, the operation of which is probably best explained by an example.

Suppose that the input pulse transformer has a 1:1 turns ratio between the primary and each half of the secondary, and that a pulse of $+2$ volts amplitude is applied to the primary. Then a pulse of $+2$ volts will appear on the upper bus, and -2 volts on the lower one. The result is to make the potential at point $A+2$ volts, while point J will be at -0.5. Then the anode of D_1 will be positive with respect to its cathode, placing the diode in a conducting state, and permitting the potential of point J to appear at point E, making the latter *negative* by 0.5 volt.

At point B, the potential is $+0.5$ volt, while at K it is $+1$, so diode D_2 is cut off, and point F will be *positive* relative to ground, although less than the potential at B by the ratio $R_6/(R_2+R_6)$. The voltages at points C and L are -1 and $+2.5$ volts respectively, so D_3 is nonconducting, and point G is negative relative to ground. The same argument may be applied to the anodes of all the diodes, and it will be found that all are negative except that of D_2. On the other hand, if the input pulse had been larger, for example, four volts, then point G would have been the positive anode, and all others negative. The net result is that for a given input level, *one and only one* diode will have an anode potential above ground. This one positive potential is amplified and made to drive the encoder.

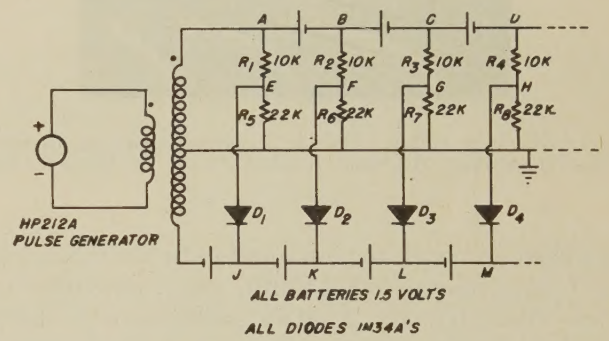


Fig. 1—Semiconductor diode quantizer.

If a given code is to be generated for an input voltage of E volts, the appropriate anode voltage must start its positive excursion at a slightly lower voltage, $E-\Delta E$, in order that it will be sufficiently positive to operate the subsequent amplifiers and encoder when the level E is reached. Likewise, if the next pulse sequence is to start for an input of E' volts, it will be found that the anode must still be slightly positive when E' is reached. Otherwise, there will be a gap when the first code is no longer produced and the second has not yet started. Thus, for a specified code to be initiated by a given diode, its anode must be positive over the range $E-\Delta E$ to $E'+\Delta E$, rather than from E to E'. The voltages E and E' are designated as "pull-in" and "drop-out" voltages in this paper. The quantity ΔE is designated as the "play" in the circuit.

Adjustments for the play of the circuit are necessary, as the amplifiers following the quantizer may have different amounts of gain from one unit to another. In practice, such adjustments are most easily made by shunting relatively low resistance potentiometers, of the order of a few hundred ohms, across the batteries in Fig. 1, and connecting points ABC, etc., and JKL, etc., to their taps. Pull-in voltages are set by taps $ABC\ldots$, and drop-out points by $JKL\ldots$. In the

* Dept. of Elec. Engrg., Purdue Univ., Lafayette, Ind.

[1] R. W. Sears, "Electron beam deflection tube for PCM," *B.S.T.J.*, vol. 27, p. 44; January, 1948.

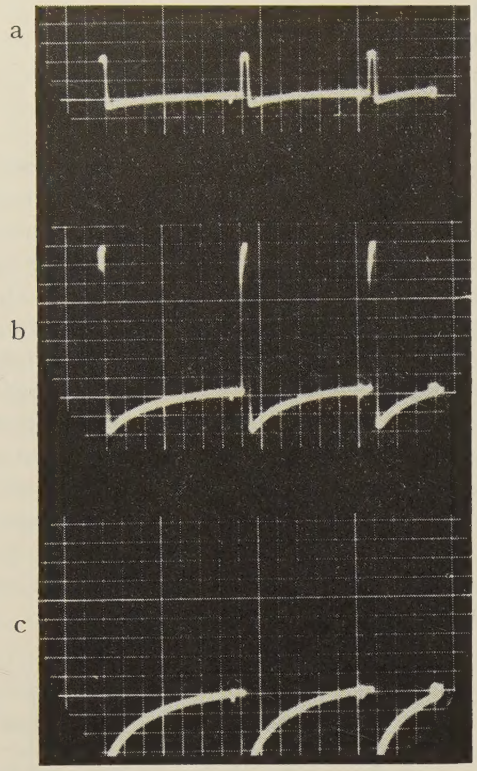


Fig. 2—Quantizer output pulses at a typical anode. (a) Input too low for encoding; (b) Input at suitable level; (c) Input too high.

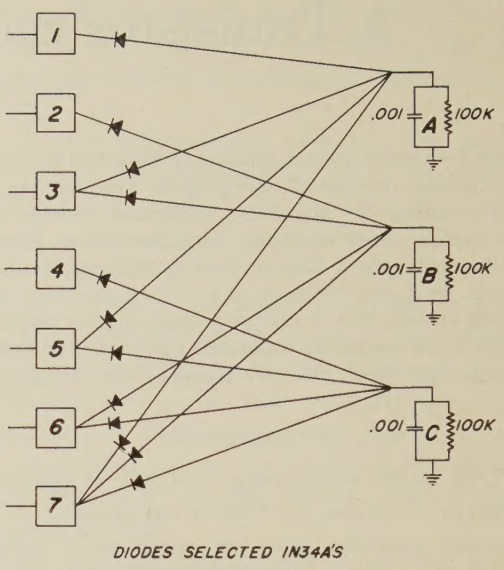


Fig. 3—The diode encoding system.

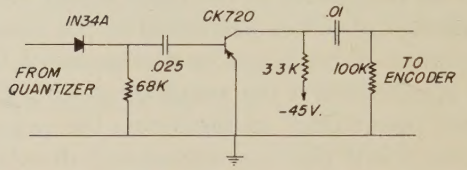


Fig. 4—Transistor amplifier driving encoder from quantizer output pulses.

experimental model, the input pulses to the transformer are of eight microseconds width, and a repetition rate of five kilocycles. Fig. 2 shows oscillograms of the potential at a representative anode for various values of input voltage.

The Encoder

The encoder circuit is adapted from switching circuits described by Brown and Rochester.[2] It is illustrated in Fig. 3, in which the numbered blocks represent the amplifiers connected between the anodes of the quantizer diodes and the encoder. (The amplifier circuit is shown in Fig. 4.) The action of this unit is more or less self-evident. For example, a positive pulse applied to the input of block 5 results in a negative pulse being transmitted to output positions *C* and *A*, giving a binary code of 101, the equivalent of a 5.

If desired, the *Gray* or *folded binary* code may be formed by reconnecting the encoder of Fig. 3 according to Table I (opposite), which includes the conventional code for comparison.

As pointed out in a recent paper by Foss, the Gray code has a number of advantages, one of the greatest being that during a change from one quantizing level to another, the maximum error that might occur is one decimal digit.[3] This advantage extends to wiring the encoder of Fig. 3 for the Gray code. For example, if the play in the system should be imperfectly adjusted so that blocks 3 and 4 are pulsed simultaneously, the conventional circuit will develop outputs on blocks *A*, *B*, and *C*, equivalent to a decimal 7. With the Gray connection, simultaneous pulsing of 3 and 4 would give outputs at *B* and *C*, corresponding to a 4. In other words, wiring for a Gray code will result in an output corresponding to *one or the other* of the two inputs excited simultaneously, rather than a serious error of many

TABLE I

Encoder Wiring Schedule for Conventional and Gray Binary Codes

Decimal Number	Conventional Code		Gray Code	
	Numbered Block	Lettered Block(s)	Numbered Block	Lettered Block(s)
1	1	*A*	1	*A*
2	2	*B*	2	*AB*
3	3	*AB*	3	*B*
4	4	*C*	4	*BC*
5	5	*AC*	5	*ABC*
6	6	*BC*	6	*AC*
7	7	*ABC*	7	*C*

[2] D. R. Brown and N. Rochester, "Rectifier networks for multiposition switching," Proc. I.R.E., vol. 37, p. 139; February, 1949.

[3] F. A. Foss, "The use of a reflected code in digital control systems," Trans.—PGEC., vol. *EC*-3, pp. 1–6; this issue.

R. E. Yaeger, "A Gray-to-binary translator and shift register," *The Transistor*, Bell Tel. Labs., Inc., New York, N. Y., pp. 611–626; 1951.

decimal digits. Thus, any concern over a moderate amount of play will be virtually eliminated.

The actual construction of this encoder was somewhat more critical than expected, since the voltages developed on the capacitors tended to leak off through the back resistances of the diodes, and charge up the remaining capacitors, giving a false code. Measurements of the back resistances of 84 1N34A diodes showed that the better half had values of 230,000 ohms or higher, and an encoder using only diodes from this half of the lot was found to be satisfactory. The problem would be aggravated with an encoder designed for four or more binary digits, as the number of diodes necessary increases rapidly with the number of bits in the code. Basically, a code of B bits will require $2^{(B-1)}B$ diodes, and although Brown and Rochester show methods of reducing this number, a 7 bit code would still require several hundred.

To assure that the encoder will start with a "clean slate" at each new 200 microsecond sampling interval, 100K resistors are shunted across the capacitors to prevent any residual code from being stored from one interval to the next. As a result of these resistors, together with the shunting effect of the back resistances of the other diodes in the matrix, the pulses appearing at the encoder outputs are of a more or less exponential shape, with a time constant appreciably shorter than would be expected from the capacitor and 100K resistor alone.

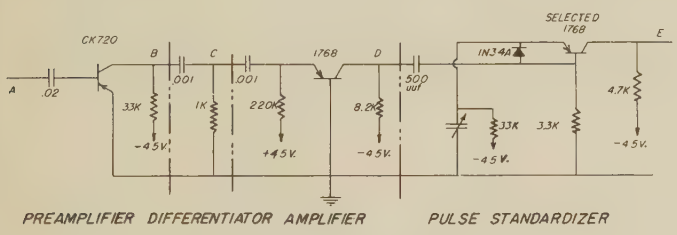


Fig. 5—Amplifier and pulse standardizing assembly joining encoder to readout system.

In order to have clean, square pulses to present to the readout system, the pulse standardizing assembly of Fig. 5 follows each of the three encoder output points. This consists of four main parts: preamplifier, differentiator, amplifier, and monostable multivibrator. The latter is a slight modification of a unit described by Lo, the 33K resistor having been found empirically to eliminate instability in the circuit.[4] The capacitor shown as variable is used to adjust the length of the output pulse, something in the neighborhood of 0.006 μf being required for an 8 microsecond pulse width. The waveforms occurring in the circuit of Fig. 5 are shown in Fig. 6. The amplitudes of these five pulses are not directly comparable, each having been photographed with a gain resulting in a convenient size trace on the oscilloscope.

[4] A. W. Lo, "Transistor trigger circuits," Proc. I.R.E., vol. 40, p. 1531; November, 1952.

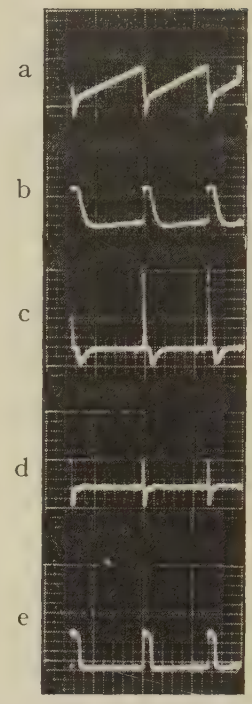


Fig. 6—Pulses in the circuit of Fig. 5. (a) Pulses at encoder output. (b) Output of preamplifier. (c) Result of differentiating pulse in (b). (d) Input to standardizer. (e) Output of standardizer. Note that letters associated with the separate parts of this figure match corresponding letters in Fig. 5.

The Readout System

The readout system consists of a set of gates and delay circuits, as shown in the block diagram of Fig. 7. Pulses from the three standardizer assemblies following the encoder are applied to one of the inputs of each of three "and" gates. The second input to each gate is the output of a delay circuit. Each of the latter, in turn, receives its input pulse from the *trailing* edge of the output of the previous delay unit, providing sequential operation of the delay circuits and therefore the gates. The input to the first delay circuit is the same as the input to the pulse transformer in Fig. 1.

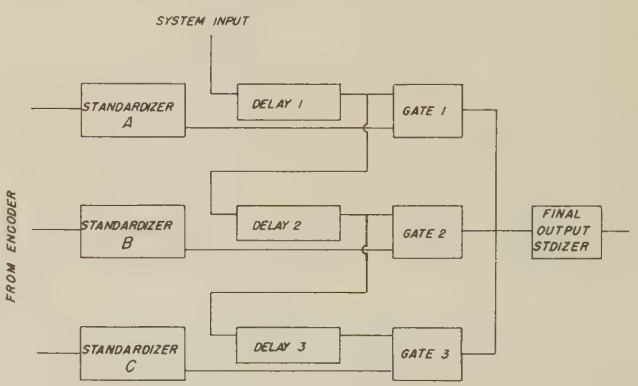


Fig. 7—Block diagram of readout system.

The events in the readout system may be best explained with Fig. 8 (next page), which shows signals at various points as functions of a common time scale. For purposes of an example, it is assumed that a 5, or

binary 101, is to be indicated at the output. Therefore, encoder outputs A and C will have negative pulses appearing simultaneously. Each of these pulses will cause the associated standardizer to produce a square wave, although these waves must be of different lengths for reasons to become apparent shortly. Both standardizer outputs rise approximately two microseconds after the appearance of the triggering pips from the encoder.

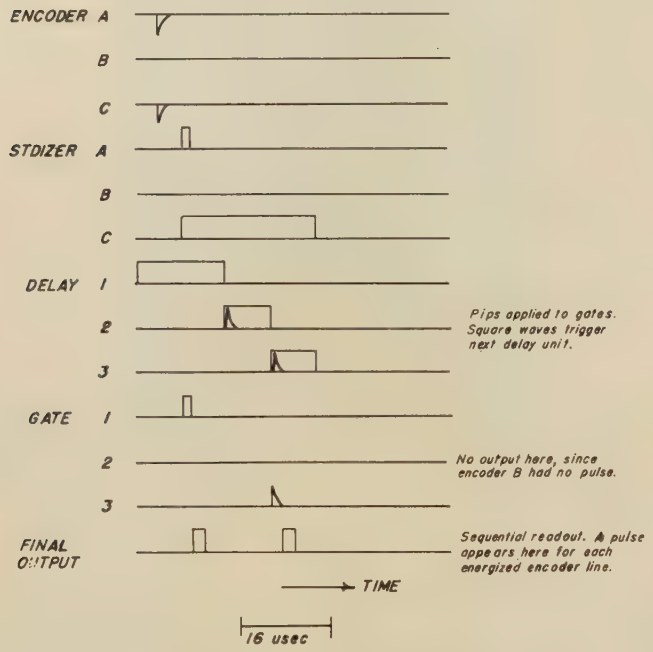


Fig. 8—Time sequence of operations during readout.

The output of delay unit 1 starts a few microseconds before the pip appears at encoder A, since it receives its triggering pulse directly from the main input for the system. It continues to remain positive for 8 microseconds after the time when output pulses first appear at the standardizers. Since the output of delay unit 1 is connected to an input of gate 1, a pulse will be delivered from gate to final output standardizer if standardizer A and delay unit 1 are simultaneously positive.

The fall of voltage when the output of delay unit 1 drops to zero produces two effects in the circuit of delay unit 2: a pip is produced almost immediately, and applied to gate 2; also, a square wave of eight microseconds length is sent to the input of delay unit 3 to postpone the latter's operation until information waiting at gate 2 may be read out. In the example of Fig. 8, there is no pulse on standardizer B, so that although the pip has enabled gate 2, there is no output from this gate.

Finally, the trailing edge of the square wave from delay unit 2 triggers delay unit 3, which provides the pip to enable gate 3 and read out the information waiting at standardizer C. It is apparent that progressively wider pulses are required from the standardizers in order that they will appear at the gates simultaneously with the enabling pips, which occur later and later for each additional binary digit in the code. The square wave output from delay unit 3 is gratuitous in this case, as it serves no purpose other than to provide the positive pip associated with its leading edge. However, it is available for adding additional delay units if the system should be expanded.

The delay units are almost identical to the standardizers, with small modifications noted below. Delay unit 1, being triggered by the main input source, needs no preamplifiers, so that connection to this unit may be made at point B in Fig. 5 with the 33K resistor and the transistor removed from the circuit.

Since a small pip is desired at the output from the gates in order to trigger subsequent circuits, it must be obtained at gate 1 by making the pulse from standardizer A very short. The enabling method used in gates 2 and 3 of applying a pip obtained from the leading edge of a delay unit will not be applicable with delay unit 1, since its leading edge rises several microseconds before any output can appear from standardizer A.

Delay units 2 and 3 are likewise modifications of a standardizer, but require the addition of a few components. In the first place, each of these units must operate on the *trailing* edge of the pulse from the output of the previous unit. This result is achieved by shunting a resistor between the base and ground of the CK720 transistor in Fig. 5. This resistor converts the preamplifier output waveform from a square wave to a sawtooth, with the *trailing edge vertical.* The value of resistance to insert at this point depends on the individual transistor with which it is associated, but a value of 220K±100K is suggested for a first trial.

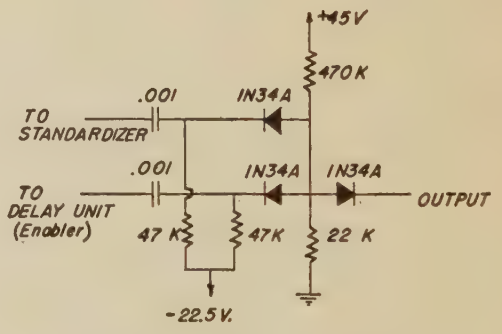


Fig. 9—Diode "and" gate.

The only other modification necessary in delay units 2 and 3 is the addition of a differentiator to provide the enabling pips for the gates. The differentiators used in this equipment consist of a 300 $\mu\mu$f capacitor in series with 10,000 ohms. They are placed between E and ground in Fig. 5, the junction between the capacitor and resistor being connected to the enabler input of the gate.

The gate circuits are all identical, and of quite conventional construction.[5] They are shown in Fig. 9.

[5] L. W. Hussey, "Semiconductor diode gates," *B.S.T.J.*, vol. 32, p. 1137; September, 1953.

The Final Output System

Even when no pulse is supposed to pass through a given gate, there may be a fairly sizeable one due to capacitive feed-through of the delay network pip, which is applied every sampling interval in any case. Such a

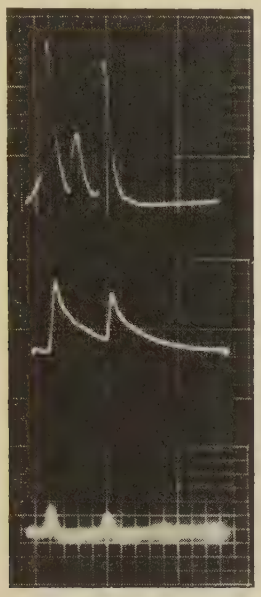


Fig. 10—Waveforms immediately following the gates. (a) Gate outputs in parallel; binary 101 encoded. (b) Spurious center pulse removed by clipping. (c) Remaining pulses differentiated to restore their form to sharp pips.

situation is shown in Fig. 10(a), which is supposed to be a binary 5, but might easily be mistaken for a 7. To eliminate this source of error, a clipper is placed after the gates, as shown in the diagram of the final output assembly in Fig. 11. The pulses remaining after the

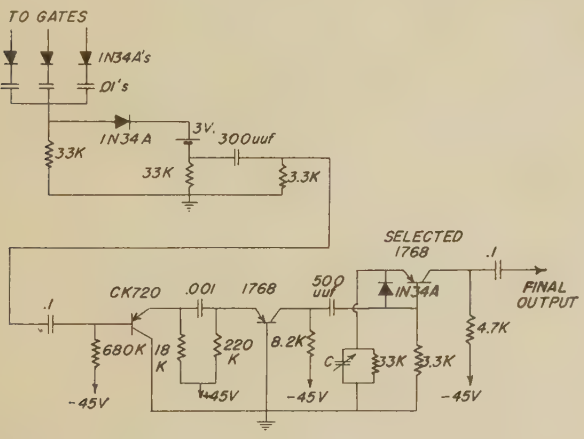


Fig. 11—Final output assembly.

clipping are seen in Fig. 10(b). Their shape leaves a great deal to be desired, so they are differentiated to restore them to sharp pips, although at considerable sacrifice in amplitude, as may be seen in Fig. 10(c). The pattern is now unmistakably a binary 5, and is ready to drive the final pulse standardizing circuits.

At this point, no more amplitude can be spared, so a high input impedance amplifier must follow the differentiator. This suggests a grounded collector amplifier, the gain of 0.8 or 0.9 in such a stage being less of a problem than the great attenuation that would occur if a relatively low impedance circuit followed the differentiator. The output of the grounded collector may be used to drive a grounded base amplifier, which in turn triggers the final pulse standardizer.

This last standardizer cannot recover from one operation in time to accept another pulse only 8 microseconds later unless the capacitor C is very small, which results in a short output pulse. This could be a disadvantage in in some cases. One remedy might be to let each gate trigger its own standardizer, in which case each pulse could be made to merge smoothly with those initiated by the preceding and following gates.

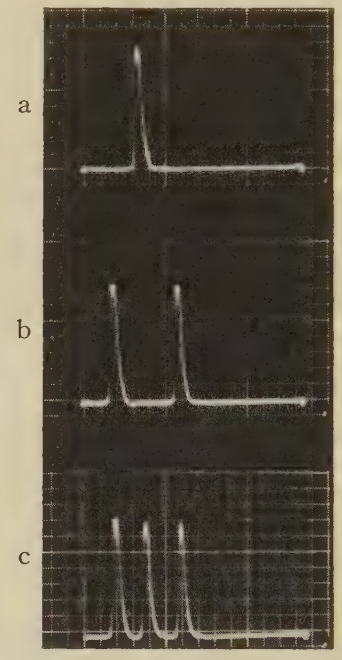


Fig. 12—Examples of output signals in binary code. Parts (a), (b), and (c) correspond respectively to decimal 2, 5, and 7.

Fig. 12 illustrates some final output pulses. They are about 2 microseconds wide, 20 volts in amplitude, and have 8 microsecond spacing between adjacent pulses. The sampling rate is 5 kc, corresponding to a set of pulses every 200 microseconds.

Conclusions

A pulse code modulator has been developed and reduced to a practical system, using only transistors, crystal diodes, resistors, and capacitors. The system accepts pulses at a rate of 5000 per second from a source, such as a Hewlett-Packard 212A pulse generator, and delivers a 3 digit binary code proportional to the amplitude of the input signal. The output pulses in each set are spaced at 8 microsecond intervals. As seems to be the

case with transistors in the present state of the art, certain units are critical, and must be pre-selected; this is especially true of the pulse standardizing networks and delay units, which operate properly with only about one-third of a typical lot of 1768 transistors. However, the encoder appears to be the limiting feature of the system, because of diode back resistance, as explained above.

To the best of the author's knowledge, the quantizer described here is novel, not having previously appeared in the literature.

ACKNOWLEDGMENT

The author wishes to express his appreciation to Dr. George R. Cooper for his original suggestion of developing a transistorized pulse code modulator and for much helpful advice during the course of the work. Thanks are also due to Mr. Paul E. Norsell who helped in the construction of much of the equipment. Financial support for this project was obtained under Contract No. DA 36-039 sc-15544 between the U. S. Army Signal Corps Engineering Laboratories and the Purdue Research Foundation.

A Radio-Frequency Nondestructive Readout For Magnetic-Core Memories*

BERNARD WIDROW†

Summary—**It is possible to read information nondestructively from two- and three-dimensional magnetic-core digital computer memories in several microseconds by exciting selected cores with rf currents. If two co-ordinate lines of a core in a memory array plane are driven at slightly different frequencies, a beat-frequency signal is generated whose phase may take on one of two values which are separated by 180 electrical degrees. These two possible phases correspond to the 0 and 1 information states of the core. The beat-frequency signal, separated from the inevitable noises by tuned linear filters, may be phase detected to yield the desired information.**

INTRODUCTION

THE SMALL ferromagnetic toroid has proven to be very effective as the basic cell in three-dimensional coincident-current memory systems for digital computers.[1,2] Its success is due to the rectangularity of its hysteresis loop, which gives it the sharp-breaking nonlinearity necessary for multico-ordinate selection and information storage.

The multico-ordinate selection scheme in general use involves destructive reading; that is, the information is destroyed when it is read out.[3,4] Additional time is therefore needed to rewrite the information thus destroyed. A high-speed, nondestructive method of reading is under development which employs radio frequency (rf) currents. Although other applications for rf readout are suggested below, the main concern of this paper is with the coincident-current memory for which this nondestructive readout is inherently suited.

Mechanics of Selection

A necessary characteristic of all linear multico-ordinate selection systems is that the individual cells to be selected must themselves be nonlinear. The excitation applied to the selected cell is a linear combination of the excitations applied to all the selecting co-ordinates, and its magnitude must be greater than the magnitude of the excitations applied to the most heavily driven nonselected cells. The effects possible at the selected cell must be very much greater (in an ideal system, infinitely greater) than the effects possible at any nonselected cell. If the cells themselves are linear, the effects at the selected cell are greater (rather than infinitely greater or very much greater) than those at any nonselected cell.

These ideas are basic to the operation of the coincident-current memory; moreover, the proposed rf readout which makes use of rf current pulses rather than dc current pulses may be evaluated in light of the above criteria.

The Coincident-Current Memory

A magnetic core has the ability to "remember" a single binary number because it may be "permanently" magnetized in either of two directions. 0 may be associated with one remanent-flux state, 1 with the other. Since this single core is to be used as a memory unit, it must be possible to write into it (magnetize it in either direction) and to read out the stored information (sense the remanent-flux direction) when desired. A practical memory would contain many such cores arranged in two- or three-dimensional arrays; the coinci-

* The research reported in this paper was supported jointly by the Army, Navy, and Air Force, under contract with Massachusetts Institute of Technology.

† Staff member, Lincoln Lab., M.I.T., Cambridge, Mass.

[1] J. W. Forrester, "Digital information storage in three dimensions using magnetic cores," *Jour. Appl. Phys.*, vol. 22, pp. 44–48; January, 1951.

[2] W. N. Papian, "A coincident-current magnetic memory cell for the storage of digital information," PROC. I.R.E., vol. 40, pp. 475–478; April, 1952.

[3] J. A. Rajchman, "A myriabit magnetic-core matrix memory," PROC. I.R.E., vol. 41, pp. 1407–1421; October, 1953.

[4] W. N. Papian, "The MIT magnetic-core memory," *IRE Proc. East. Joint Computer Conf.*, pp. 37–42; December, 1953.

dent-current scheme allows individual cores to be selected so that they may be written into and read out of.

Consider the two-dimensional array of cores shown in Fig. 1. Let each core have the dc hysteresis loop shown in Fig. 2. If the drivers are able to supply X and Y currents of magnitude $I_m/2$, any core in the array can be set in the 0 or 1 state by coincidentally exciting the corresponding X and Y lines (single-turn exciting windings) with currents of the proper polarity for a time long enough to permit flux switching. Only the selected core receives the full I_m; the half-selected cores receive $I_m/2$ which is of too small a magnitude to cause flux switching and therefore may disturb but will not destroy the stored information.

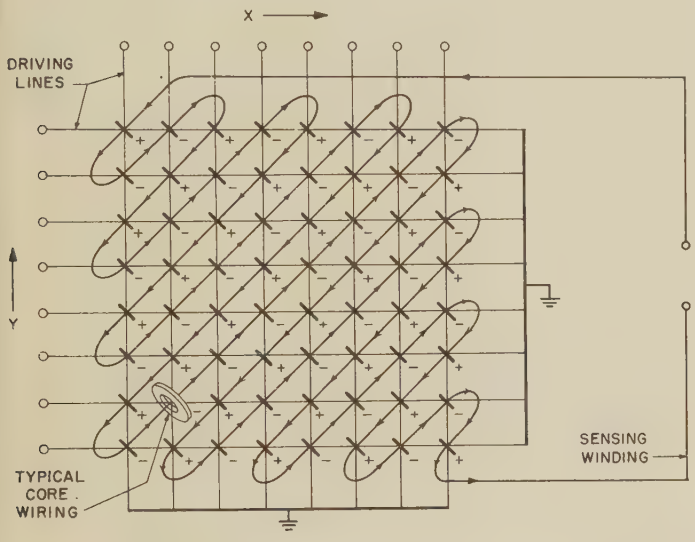


Fig. 1—8×8 model of a memory array plane.

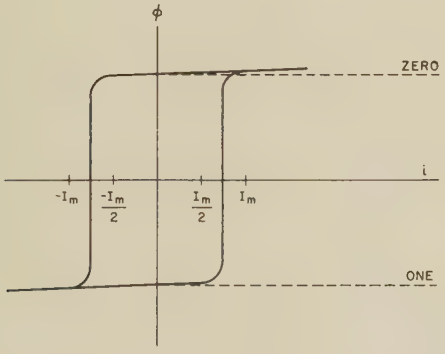


Fig. 2—D-C hysteresis loop of a typical magnetic core.

The use of coincident currents for reading is similar to their use for writing. In order to sense the contents of a core in a memory plane, 0 is written into it. If the selected core already contains a 0, the 0 state remains, and only very small flux changes take place. If, on the other hand, the selected core contains a 1, full flux reversal will occur. A sensing winding (see Fig. 1) threaded through every core in the array will exhibit either a small noise pulse corresponding to a 0 in the selected core or a much larger and longer voltage pulse signifying that the selected core had contained a 1. This pulse readout is destructive, and provision must be made (and time allowed) for rewriting if the information is to be retained.

The signal induced in a sensing winding during readout is not generated by the selected core alone. In a 4×4 planar array of cores, there are always six half-selected cores producing noise when a selected core is pulsed. The sensing winding passes through adjacent cores on a common X or Y driving line in the alternating fashion shown in Fig. 1. The intention is that half-selected noises should cancel as they would if the half-selected outputs were all identical. They are not identical because of variations in properties among the cores; more significant, however, is the difference in the noise outputs of a core containing a 1 and of the same core storing a 0. The resultant of these noises is proportional to n for an $n \times n$ array, while the signal size is fixed. Signal-to-noise ratios are inversely proportional to n for a given core type.

Nondestructive RF Readout

At the expense of more elaborate driving circuits, an rf readout system is obtainable which could be fast (speed determined mainly by frequencies of rf driving currents; information is read nondestructively, so that time for rewriting need not be spent), and reliable in large systems (signals separated from the inevitable noises by tuned filters).

A Two-Dimensional RF Readout System

If the X and Y lines corresponding to a chosen core in a memory plane (Fig. 1) are driven simultaneously by two sinusoidal rf currents of frequencies ω_1 and ω_2, these currents add at the selected core (a nonlinear device) and cause it to induce a difference-frequency beat in the sensing winding. The frequencies of the two driving currents are made to be much greater than their difference so that the difference-frequency signal may be easily separated from the high-frequency noises.

According to the selection criteria, rf readout gives an ideal type of selection. The only cell in the two-dimensional system able to generate a difference-frequency signal is that one at which the two rf currents are superposed within a nonlinear magnetic circuit. The half-selected cores (those sharing a single X line or a single Y line with the selected core) produce only noises at the fundamental drive frequencies and at their higher-order harmonics. These noises are linearly superposed in the sensing winding and may be removed by linear passive filters.

It is possible to detect the remanent-flux state of the selected core from the phase of the difference-frequency signal; this phase may assume one of two values (separated from each other by 180 degrees) corresponding to a 1 or a 0 stored in the selected core. To show that a difference-frequency signal is so generated and to predict the operation of a core in a memory plane during

rf readout, a single core rf pulse tester was constructed (see Fig. 3). (Since a trigger pulse starts both oscillators, the two rf currents and the difference-frequency voltage from the tuned amplifier are locked in on the same sweep.) As predicted, polarity of the difference-frequency signal reversed with the core's remanent flux.

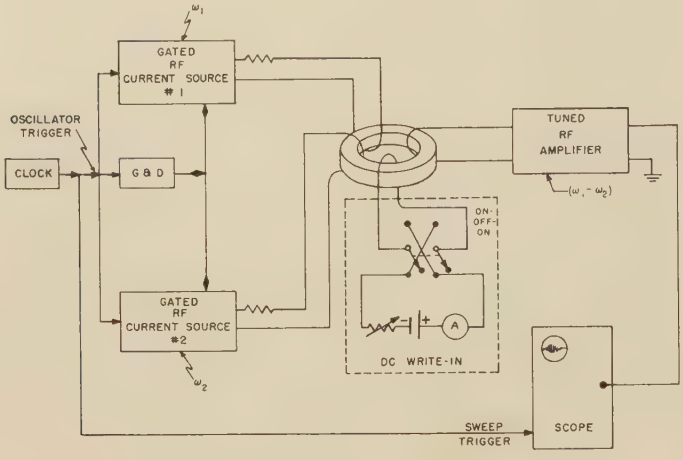


Fig. 3—RF pulse tester.

Analysis of Beat Generation

Assume that the two rf hysteresis loops of the two memory states are the single-valued nonlinear analytic $\phi - i$ relations of Fig. 4, and that the type of symmetry they show to each other is the same as the symmetry between the top (0 state) and bottom (1 state) of the dc hysteresis loop of Fig. 2. The relationship between

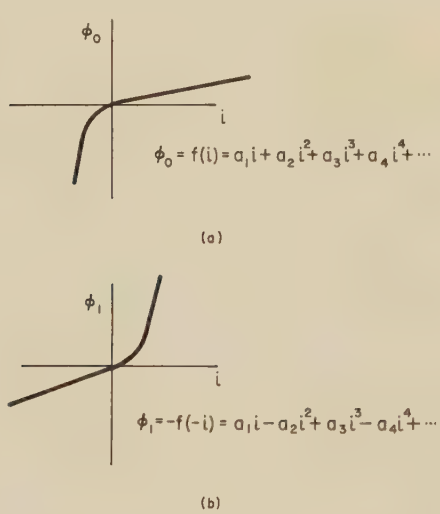


Fig. 4—Incremental flux vs instantaneous current when core stores 1 and 0.

ϕ_0 and ϕ_1 of Fig. 4 becomes apparent when it is observed that either curve may be made identical to the other by a "double flip" which reverses both the flux and current scales. If the 0 state incremental flux $\phi_0 = f(i)$, then the 1 state incremental flux $\phi_1 = -f(-i)$. Let the incremental fluxes be expanded in a Taylor's Series in the instantaneous current i:

$$\phi_0 = f(i) = a_1 i + a_2 i^2 + a_3 i^3 + \cdots \quad (1)$$

$$\phi_1 = -f(-i) = a_1 i - a_2 i^2 + a_3 i^3 - \cdots \quad (2)$$

To generate beats, let the two sinusoidal currents flow simultaneously. Then

$$i = (\sin \omega_1 t + \sin \omega_2 t) \quad (3)$$

$$\phi_0 = a_1(\sin \omega_1 t + \sin \omega_2 t) + a_2(\sin \omega_1 t + \sin \omega_2 t)^2 + \cdots \quad (4)$$

$$\phi_1 = a_1(\sin \omega_1 t + \sin \omega_2 t) - a_2(\sin \omega_1 t + \sin \omega_2 t)^2 + \cdots . \quad (5)$$

The first terms of (4) and (5) cause outputs at the two fundamental driving frequencies. The second terms are of the form

$$a_2(\sin \omega_1 t + \sin \omega_2 t)^2 = a_2(\sin^2 \omega_1 t + \sin^2 \omega_2 t + 2 \sin \omega_1 t \sin \omega_2 t). \quad (6)$$

Outputs at the second-harmonic frequencies of the fundamental driving frequencies are caused by the terms $a_2 \sin^2 \omega_1 t$ and $a_2 \sin^2 \omega_2 t$. The remaining component is:

$$2a_2 \sin \omega_1 t \sin \omega_2 t = a_2 \cos (\omega_1 - \omega_2)t - a_2 \cos (\omega_1 + \omega_2)t. \quad (7)$$

This gives a sum-frequency signal and the desired difference-frequency signal.

The squaring arising from the nonlinear addition of two sinusoids creates the difference-frequency-flux component. Notice that the sign of the coefficient a_2 is reversed as the core is switched. Physically, this means that the curvature of the ϕ-versus-i path is positive or negative, depending on whether the core stores a 1 or a 0. It then follows that the polarity of the difference-frequency signal induced in a sensing winding reverses when the core's information content is changed.

The above shows that core properties are not critical. Different cores may exhibit different values of the coefficient a_2. Although this would cause them to generate beats whose amplitudes may differ, their phases must be discrete. This has been verified experimentally for cores of many sizes and materials. The perfection of the polarity reversal has also been checked to within a few degrees (this check was limited by scope accuracy) for every core tried.

Why RF Readout Is Nondestructive

The periods of the rf waves are very much shorter than the response times of the cores for the current magnitudes involved. The same internal mechanisms that limit the switching speed of a core driven by dc pulses are believed to prevent the destruction of information by rf currents even when the zero-to-peak of the rf excitation is made many times greater than the coercive force. If any flux switching is done during a half cycle, it is immediately undone during the next half cycle. If a small dc bias is applied, such that the bias plus the zero-to-peak of the rf excitation exceeds the coercive force, then the core will slowly switch in the

direction of the bias. In the absence of dc bias, a core may be rf driven indefinitely and experience only negligible deterioration of its remanent flux.

Conclusion

An experimental 16×16 memory plane was constructed of steel-ribbon cores having the following specifications: 10 wraps of 4-79 molybdenum Permalloy-ribbon, $\frac{1}{8}$-mil thick, $\frac{1}{8}$-inch wide, wound on a $\frac{1}{8}$-inch bobbin. These cores had been rejected for use in the usual coincident-current memory because of their nonuniformities. They had shown variations of ten to one among some of their pulse readout signals.

It was possible to sense any core in this array plane using the rf readout. X and Y coordinate lines were manually selected by clip-lead connection to two pulsed rf current sources which supplied currents at 5.4 and 6.8 megacycles respectively. (Fig. 5 shows an rf readout system for a 4×4 memory.)

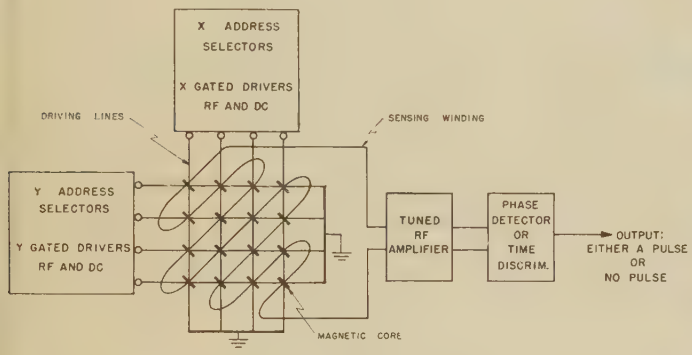


Fig. 5—An rf readout system.

Typical driving-current and sensing-amplifier waveforms are shown in Fig. 6. The frequencies chosen were arbitrary but convenient for the available equipment. The desired information is discernible about 2 microseconds after the beginning of the reading operation: the rf output of the sensing amplifier requires about 15 microseconds to build up. If the rf readout is to be used to read information from random addresses in high-speed succession, transient recovery time must be sharply reduced from that shown in Fig. 6. An obvious solution is reduction of the driving time of the rf current sources from the 15 microseconds of Fig. 6 (for the sake of illustration, made longer than necessary) to about 1 or 2 microseconds. The bandwidth of the sensing amplifier might be increased for faster rf transients. (The amplifier used had a bandwidth of 75 kilocycles.) To achieve the same noise rejection with the tuned sensing amplifier, it would be necessary to raise the driving frequencies. It would also be possible to use an electronic damper in the sensing amplifier that could be gated on as soon as the desired information is obtained.

A practical memory would use its electronic address-selection and current-driver circuits for both the rf readout and the usual pulsed write-in. Although such circuits have not yet been tried, experiments made show that rf readout could be used in large three-dimensional memories. Other applications of rf driven cores seem promising. A magnetic-core may be used as a low impedance mixer in communications circuits by applying two or possibly several inputs to separate primary windings. The mixed output is induced in a secondary winding. Magnetic cores may also be used in carrier-type logical circuits, where advantage may be taken of their ability to detect coincidences (the generation of beat-frequency signal depends upon the presence of two or more inputs), and where the necessary information storage could be accomplished by the cores themselves. The prime advantage of magnetic cores for these mixing applications lies in the *long-term stability* of their nonlinear knee, as contrasted to the short-term stability of vacuum-tube mixers.

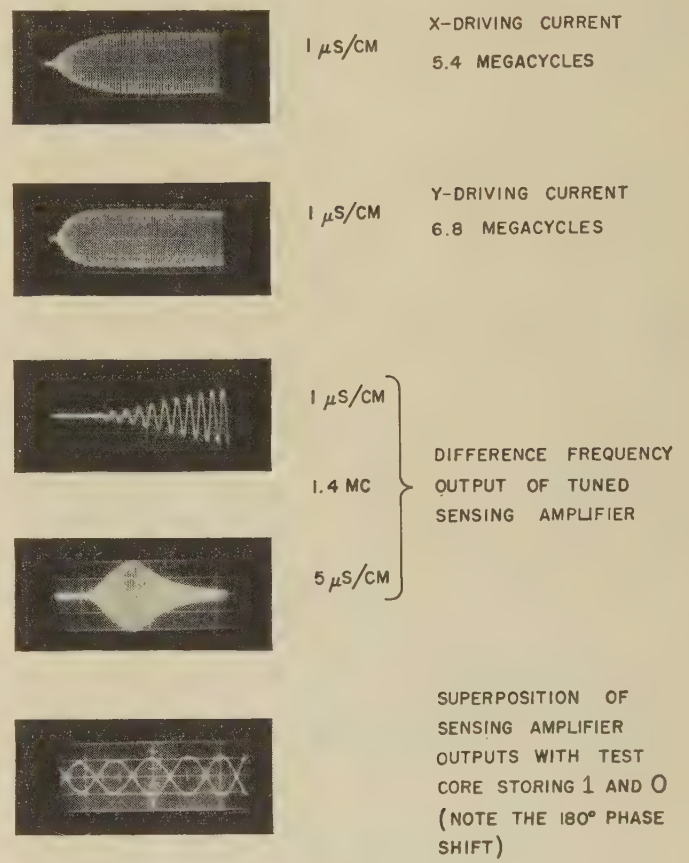


Fig. 6—Typical driving-current and sensing-amplifier wave forms.

Acknowledgment

The author wishes to thank Professor W. K. Linvill of the Electrical Engineering Department of M.I.T. for his assistance and supervision of this research. He also wishes to acknowledge the help of Messrs. W. N. Papian, J. B. Bennett, and D. A. Buck of the Lincoln Laboratory staff in editing this paper.

Time-Delay Networks For an Analog Computer*

W. J. CUNNINGHAM†

Summary—Time-delay networks suitable for an analog computer are designed by considering the location of poles and zeros in their transfer functions. The curve of phase shift against frequency should be a straight line. The negative slope of this curve is the time delay. Even-order derivatives of the curve automatically vanish at zero frequency. Roots of the transfer function are chosen to make vanish similarly as many as possible of the derivatives of odd order, higher than the first. Data are given for networks with one, two, three, and four pairs of roots.

THERE has appeared recently a design for a time-delay network for use with an analog computer.[1] The design was based on the procedure of expressing the delay operator $\exp(-sT)$ in a special fractional form, and synthesizing a circuit to represent this fraction. The purpose of the present discussion is to point out a somewhat different approach to the same problem. The resulting networks are quite similar.

The transfer function for a network can be written as

$$H = E_2(s)/E_1(s) \tag{1}$$

where E_2 and E_1 are the complex signals at the output and input terminals respectively, and s is the complex frequency variable. For a time-delay network, the magnitude of H should be unity and its angle should vary linearly with real frequency. The variation of H with s can be studied most easily by considering the roots of the polynomials in s that form the numerator and denominator of the fraction representing H. It has been shown that the roots must fit a specified pattern.[2] They must occur as real quantities or as complex conjugate pairs. Roots of the denominator, or poles, must occur in the left half of the complex-frequency plane, and roots of the numerator, or zeros, must occur in the right half-plane. Each pole must have a matching zero with its real part of opposite algebraic sign. A typical situation is shown in Fig. 1, where six roots appear. The design problem is essentially that of determining the optimum locations for these roots.

As an example, a network with the six roots of Fig. 1 is used. Its transfer function is:

$$H = \frac{[s-(a+jb)][s-(a-jb)][s-c]}{[s+(a-jb)][s+(a+jb)][s+c]}. \tag{2}$$

For real driving frequency, $s=j\omega$ and with definitions $x \equiv b/a$, $y \equiv c/a$, $\Omega = \omega/a$, the angle of H is:

$$\theta = 2\left[\tan^{-1}(x-\Omega) - \tan^{-1}(x+\Omega) - \tan^{-1}(\Omega/y)\right]. \tag{3}$$

The time delay for the network is

$$T = -d\vartheta/d\omega = -(1/a)d\vartheta/d\Omega. \tag{4}$$

If T is to be constant, the phase curve must be linear with respect to frequency, thus having the same slope at all frequencies. The actual curve for a finite network departs from linearity, so that an initial delay T_0, existing at zero frequency, can be defined as

$$T_0 = -(1/a)D\theta \tag{5}$$

where $D\theta$ represents $d\theta/d\Omega$ evaluated at $\Omega=0$. In general, $D^n\theta \equiv d^n\theta/d\Omega^n$, also evaluated at $\Omega=0$.

For the phase curve to be linear, all its derivatives higher than the first should vanish. Again, this criterion can be satisfied at only one frequency, most conveniently chosen at $\Omega=0$. All even-order derivatives vanish automatically at $\Omega=0$ because the phase curve is an odd function of frequency. The parameters a and b remain to be chosen, and this choice is made so as to give $D^3\theta=0$ and $D^5\theta=0$.

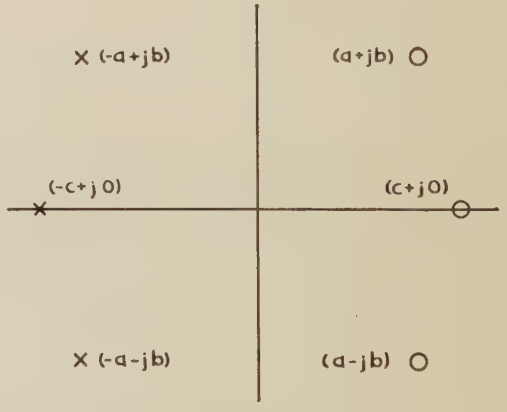


Fig. 1—Location of poles and zeros of transfer function for 6-root time-delay network.

The derivatives of interest here are as follows.

$$D\theta = -2\left[\frac{2}{1+x^2} + \frac{1}{y}\right] \tag{6}$$

$$D^3\theta = 4\left[\frac{2(1-3x^2)}{(1+x^2)^3} + \frac{1}{y^3}\right] \tag{7}$$

$$D^5\theta = -48\left[\frac{2(1-10x^2+5x^4)}{(1+x^2)^5} + \frac{1}{y^5}\right]. \tag{8}$$

These last two derivatives can be set to zero, and the resulting simultaneous equations solved for x and y.

* The research reported in this paper was done in part under Contract Nonr—433(00), between Yale University and the Office of Naval Research.

† Dept. of Elec. Engrg., Yale University, New Haven, Conn.

[1] C. D. Morrill, "A sub-audio time delay circuit," TRANS. IRE, PGEC, vol. 3, pp. 45–49; June, 1954.

[2] H. W. Bode, "Network Analysis and Feedback Amplifier Design," D. Van Nostrand Company, Inc., New York, N. Y., p. 239; 1945.

This leads to the numerical values $x = 0.953$ and $y = 1.26$. The initial time delay is, from (5) and (6),

$$T_0 = 3.68/a. \tag{9}$$

By putting the values for x and y into (3), data for a curve of θ as a function of Ω can be calculated. This curve is shown in Fig. 2, along with a straight line having the slope found from (6).

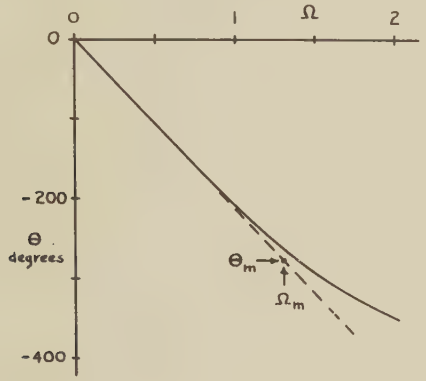


Fig. 2—Phase shift of 6-root delay network as a function of normalized frequency, $\Omega = \omega/a$. At Ω_m the actual phase angle is 5 per cent smaller than for the straight line corresponding to the same initial delay.

A maximum frequency limit can be set by assigning an allowable departure of the phase curve from the straight line. If five per cent is arbitrarily set for this departure, the maximum value of Ω is $\Omega_m = 1.3$, or the maximum angular frequency is

$$\omega_m = 1.3a. \tag{10}$$

The product, $\omega_m T_0$, is a sort of figure of merit, $\omega_m T_0 = 4.8$. This figure of merit expresses in radians the maximum phase shift that can be obtained within the allowed five per cent departure. Thus, an alternate form for the figure of merit is $\theta_m = 277$ degrees, which is merely $\omega_m T_0$ expressed in degrees. The actual phase shift at ω_m is five per cent less than this value of θ_m.

By this same process, networks with other numbers of roots can be studied. If the number of pairs of roots is odd, one pair must be located on the real axis; if the number of pairs is even, all are located off the axis. As the number of roots is increased, the algebra needed to determine their optimum location becomes increasingly complicated.

Tables I–IV (following page) give results of analyses of this sort for networks having one, two, three, and four pairs of roots. Also shown in the tables are the combinations of computer elements that can be used to provide the necessary transfer functions. One circuit can be used to give a single pair of roots; another circuit provides two pairs. These two circuits can be arranged in tandem to give any larger number of pairs of roots. It is possible to devise alternate configurations to provide the larger numbers of roots. Such alternate arrangements that have been explored, however, require at least as many elements as the combination of several of the simpler circuits. A combination of simple circuits has the advantage that it can easily be broken down for test purposes, should this be desirable.

In choosing a network for a given application, the primary consideration is the maximum phase shift that must be obtained with good linearity of the phase curve. If the maximum shift need be only small, a simple network is adequate; if the maximum shift is large, a complicated network is required. The maximum shift obtainable is given by the figure of merit, $\omega_m T_0$ or θ_m, listed in the tables.

It is worth noting that two identical networks in tandem provide double the initial time delay of either one alone, but that the maximum frequency is unchanged. The figure of merit for the combination of two networks is twice that of a single one. The same circuit elements readjusted to use the available roots most effectively will give a larger figure of merit. Thus, for two identical 4-root networks, $\theta_m = 2 \times 155 = 310$ degrees, while the same elements adjusted to give one 8-root network yield $\theta_m = 470$ degrees. It is generally advantageous to use a network based on a single design with sufficient roots to give the necessary total phase shift.

One application of these delay networks has been in studying a nonlinear differential-difference equation of growth:

$$dX(t)/dt = AX(t) - BX(t)X(t - T), \tag{11}$$

where A, B, and T are all positive real constants.[3] From a preliminary analysis it was known that, with product AT large enough, solution for X is a periodic oscillation of relaxation type, containing a number of harmonics of the fundamental frequency. The period of the oscillation is in the order of five times the delay time, T. Thus, at the fundamental frequency the phase shift is around 360/5, or about 75 degrees. The phase shift for a harmonic component is the order of harmonic multiplied by the shift of the fundamental. If six harmonics are of importance here, the total phase shift must be $6 \times 75 = 450$ degrees. An 8-root delay network is indicated.

A delay time, T_0, of two seconds was chosen as convenient for the computer being used. The design data for the delay network are as follows.

$$a = 5.28/T_0 = 2.64 \text{ sec}^{-1}$$
$$b = xa = .34 \times 2.64 = 0.895 \text{ sec}^{-1}$$
$$c = ya = .71 \times 2.64 = 1.87 \text{ sec}^{-1}$$
$$d = zc = 1.5 \times 1.87 = 2.81 \text{ sec}^{-1}$$
$$a^2 + b^2 = 7.72 \text{ sec}^{-2}$$
$$c^2 + d^2 = 11.41 \text{ sec}^{-2}.$$

A network based on these parameters was found to lead to solutions of good accuracy.

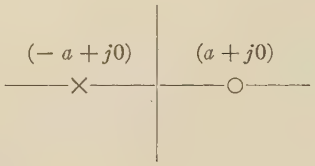
[3] W. J. Cunningham, A Nonlinear Differential-Difference Equation of Growth, Contract Nonr-433(00), Yale University, May 1954; W. J. Cunningham, *Proc. Nat. Acad. of Sci.*, vol. 40, p. 708; August, 1954.

TABLE I

2-Root Network

Pairs of roots: one

Location:



Design criteria: $D^2\theta = 0$

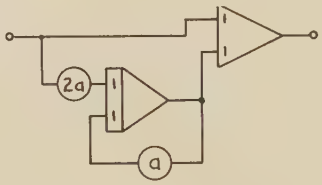
Circuit parameters: only a is specified

Initial delay: $T_0 = 2/a$

Maximum frequency: $\omega_m = 0.4a$

Figure of merit: $\omega_m T_0 = 0.8$, $\theta_m = 46$ degrees

Circuit:



Transfer function:

$$H = -\frac{(s-a)}{(s+a)}.$$

TABLE II

4-Root Network

Pairs of roots: two

Location:

(− a + jb) ×	(a + jb) ○
× (− a − jb)	○ (a − jb)

Design criteria: $D^2\theta = D^3\theta = D^4\theta = 0$

for $D^3\theta = 0$: $(1 - 3x^2) = 0$

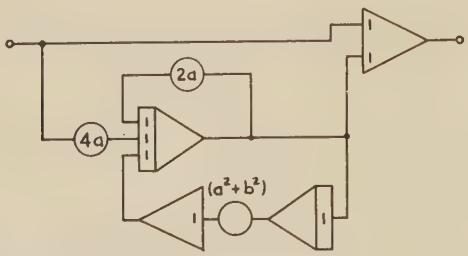
Circuit parameters: $x = b/a = 0.577$

Initial delay: $T_0 = 3/a$

Maximum frequency: $\omega_m = 0.9a$

Figure of merit: $\omega_m T_0 = 2.7$, $\theta_m = 155$ degrees

Circuit:



Transfer function:

$$H = -\frac{[s^2 - 2as + (a^2+b^2)]}{[s^2 + 2as + (a^2+b^2)]}.$$

TABLE III

6-Root Network

Pairs of roots: three

Location:

(− a + jb) ×	○ (a + jb)
(− c + j0) ×	○ (c + j0)
(− a − jb) ×	○ (a − jb)

Design criteria: $D^2\theta = D^3\theta = D^4\theta = D^5\theta = D^6\theta = 0$

for $D^3\theta = 0$: $\dfrac{2(1-3x^2)}{(1+x^2)^3} + \dfrac{1}{y^3} = 0$

for $D^5\theta = 0$: $\dfrac{2(1-10x^2+5x^4)}{(1+x^2)^5} + \dfrac{1}{y^5} = 0$

Circuit parameters: $x = b/a = 0.953$

$y = c/a = 1.26$

Initial delay: $T_0 = 3.68/a$

Maximum frequency: $\omega_m = 1.3a$

Figure of merit: $\omega_m T_0 = 4.8$, $\theta_m = 277$ degrees

Circuit: The circuit of Table I, designed with parameter c, followed by the circuit of Table II, designed with parameters a and b.

Transfer function:

$$H = \frac{[s^2 - 2as + (a^2+b^2)][s-c]}{[s^2 + 2as + (a^2+b^2)][s+c]}.$$

TABLE IV

8-Root Network

Pairs of roots: four

Location:

(− c + jd) ×	○ (c + jd)
(− a + jb) ×	○ (a + jb)
(− a − jb) ×	○ (a − jb)
(− c − jd) ×	○ (c − jd)

Design criteria: $D^2\theta = D^3\theta = D^4\theta = D^5\theta = D^6\theta = D^7\theta = D^8\theta = 0$

for $D^3\theta = 0$: $\dfrac{1-3x^2}{(1+x^2)^3} + \dfrac{1-3z^2}{y^3(1+z^2)^3} = 0$

for $D^5\theta = 0$: $\dfrac{1-10x^2+5x^4}{(1+x^2)^5} + \dfrac{1-10z^2+5z^4}{y^5(1+z^2)^5} = 0$

for $D^7\theta = 0$: $\dfrac{1-21x^2+35x^4-7x^6}{(1+x^2)^7} + \dfrac{1-21z^2+35z^4-7z^6}{y^7(1+z^2)^7} = 0.$

Circuit parameters: $x = b/a = 0.34$

$y = c/a = 0.71$

$z = d/c = 1.5$

Initial delay: $T_0 = 5.28/a$

Maximum frequency: $\omega_m = 1.55a$

Figure of merit: $\omega_m T_0 = 8.2$, $\theta_m = 470$ degrees

Circuit: The circuit of Table II, designed with parameters a and b, followed by a second circuit of Table II, designed with parameters c and d.

Transfer function:

$$H = \frac{[s^2 - 2as + (a^2+b^2)][s^2 - 2cs + (c^2+d^2)]}{[s^2 + 2as + (a^2+b^2)][s^2 + 2cs + (c^2+d^2)]}.$$

A Stabilized Driftless Analog Integrator

HOWARD HAMER*

Summary—A new chopper-stabilized analog integrator circui, has been developed which, at the cost of two passive elementts enables the stabilizer amplifier to operate on the output drift due to input current as well as the drift due to unbalance.

Introduction

ONE OF the most important operational components used in electronic analog computation is the electronic integrator, which, in its most common form, consists of a high-gain direct-coupled amplifier with negative feedback through a capacitor. Unless special precautions are taken, such an integrator will be subject to output drift, which may be defined for this purpose as any change in output voltage not caused by an applied input voltage. This drift might be due to many factors, the most important and common of which are amplifier unbalance and input current. If chopper-stabilization is added to the amplifier, as is standard in precision computing systems, the component of drift due to unbalance is reduced by the gain of the stabilizing amplifier, as will be demonstrated below. However, the drift due to input current, which is normally the second largest source, is not reduced, as will also be shown, by addition of a stabilizing amplifier.

A new chopper-stabilized integrator circuit has been developed which reduces the drift due to all active components, such that the resulting residual drift is due only to the unbalance of a balanced bridge, which consists of passive elements of reasonable size.

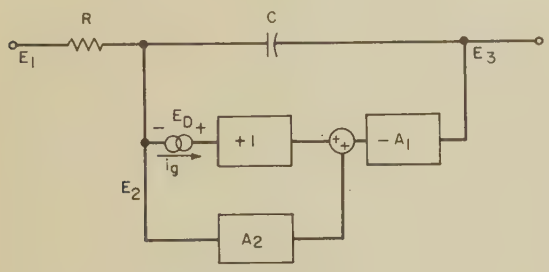


Fig. 1—Standard integrator.

Conventional Drift Reduction Techniques

The conventional chopper-stabilized integrator circuit is shown schematically in Fig. 1 above. In this diagram, E_1 is the input signal in Laplace transform form, E_2 is the signal at the input to the high-gain amplifier, and E_3 the signal at the output. R is the input resistor and C the feedback capacitor. E_D represents the amplifier unbalance referred to the input, and i_g represents the input current. The blocks marked $+1$ and $-A_1$ combine to form the main amplifier, and the block marked A_2 represents the stabilizing amplifier. The output voltage, in terms of the inputs to the amplifier, is:

$$E_3 = -A_1(E_2 + E_D + A_2E_2) \tag{1}$$

and the voltage at the input to the amplifier is

$$E_2 = \frac{E_1 + TsE_3 - i_gR}{1 + Ts} \tag{2}$$

where $T = RC$ and s = the Laplacian operator. Combining (1) and (2)

$$E_3 = -A_1(1 + A_2)\left(\frac{E_1}{1 + Ts} + \frac{TsE_3}{1 + Ts} - \frac{i_gR}{1 + Ts}\right) - A_1E_D \tag{3}$$

or

$$E_3 = -\frac{E_1}{Ts} + \frac{i_g}{Cs} - \frac{E_D}{(1 + A_2)Ts} - \frac{E_D}{1 + A_2}$$

if

$$A_1A_2 \gg 1.$$

Eq. 3 shows the output voltage approximately the negative integral of the input, plus integral of the input current divided by C, plus some drift terms, in which effect of drift voltage is divided by the gain of the stabilizing amplifier, as was indicated in the Introduction.

There are three methods which have been most commonly used to reduce the drift due to input current. One method is to increase the value of C. However, this capacitor must be of the finest quality, and high quality precision capacitors of large capacity are extremely expensive. A second method is to capacitively couple the input to the main amplifier, allowing the chopper amplifier to maintain response at dc and low frequencies, and thus blocking out the input current which is essentially dc. This arrangement, however, suffers in that the input condenser can be easily charged through the input tube, thus blocking the amplifier. The third method of drift reduction uses a bleeder resistance from the main amplifier input to a power supply. This, in effect, supplies the input current from a source other than the integrating condenser, and thus reduces drift due to input current. This circuit is useful as long as the input current remains constant, but in most cases this current is continuously changing, and the bleeder resistor would have to be varied to match these changes in input current.

* Electronic Associates, Inc., Long Branch, N. J.

Drift Reduction by Bridge Balance

Fig. 2 shows a circuit which will reduce the drift in an integrator output due to input current, using the same active components as does the standard integrator, with the addition of passive elements. A second resistor and capacitor are added, the product of which is equal to the

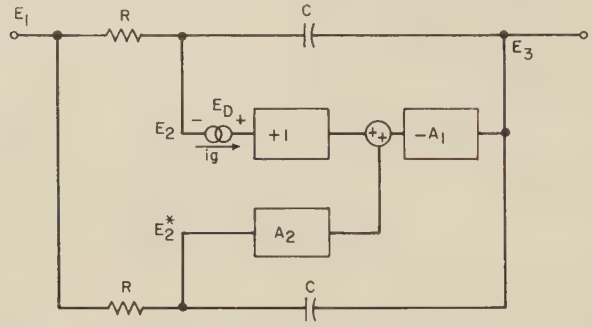


Fig. 2—Compensated integrator.

product of the original resistor and capacitor. The chopper input is taken from the junction of these new elements, which are connected in a bridge circuit with the original elements. The output voltage, in terms of the inputs to the amplifier, is now

$$E_3 = -A_1(E_2 + E_D + A_2E_2^*), \tag{4}$$

the voltage at the input to the main amplifier is

$$E_2 = (E_1 + TsE_3 - i_gR)/(1 + Ts), \tag{5}$$

and the voltage at the input to the chopper is

$$E_2^* = \frac{E_1 + TsE_3}{1 + Ts}. \tag{6}$$

Combining (4) to (6), and making the same approximation as before,

$$E_3 = -\frac{E_1}{Ts} + \frac{i_g}{(1 + A_2)Cs} - \frac{E_D}{(1 + A_2)Ts} - \frac{E_D}{1 + A_2}. \tag{7}$$

Comparing (7) with (3), we see that the new circuit has reduced the rate of change of output voltage due to input current by the gain of the stabilization amplifier.

For experimental verification of the above conclusion, an amplifier was chosen on an Electronic Associates, Inc. Analogue Computer which drifted, when connected as a standard integrator, 10 mv in 104 seconds. The new circuit was patched in, and the maximum drift observed in the same time was 0.4 mv. No care was taken to match the time constants of the two networks to better than standard tolerance. It is expected that if the time constants were matched accurately, the drift rate would be reduced even more.

A Desk-Model Electronic Analog Computer

H. A. ROSEN* and M. W. FOSSIER*

Summary—**A description is given of an analog computer comparable in size to an automatic desk calculator. By combining the techniques of electronic and electric analogs, a high problem-solving capacity is achieved in a minimum of space. Nine stable high-gain dc amplifiers are provided, each of which can be made to produce a variety of functional responses. This is accomplished by allowing the operator to build up the input and feedback networks of each amplifier on a plug board, on which space is provided for additional networks involving resistors, capacitors, and crystal diodes. The plug board containing all the components and interconnections involved in a given problem can be stored in a filing cabinet when not in use. The low cost and small size of the computer make it suited for use in many situations in which an analog computer would not otherwise be used.**

Introduction

IN RECENT years there has been a rapid growth in the use of electronic analog computers, particularly in the design of automatic control systems for piloted and pilotless aircraft. The use of these computers has been primarily limited to large and costly installations manned by a permanent staff. The principal advantage of this type of installation lies in its ability to solve highly complex problems. However, this is offset to some extent by the fact that the computer operators are often unfamiliar with the problem under investigation, and therefore unable to distinguish between adequate and superfluous simulation. This situation has resulted many times in an analog more complex than is justified by the problem. A separate computer facility also often requires the inconvenience of scheduling time on the computer considerably in advance of the date desired.

The computer described in the present paper was developed to provide the design engineer with a compact low-cost computer capable of solving many of the problems arising in the design of aircraft guidance and control systems. Because of the low cost, it is possible to provide a number of computers in an engineering department, much as is done with arithmetic calculators,

* Missile and Radar Division, Raytheon Mfg. Co., Oxnard, Calif.

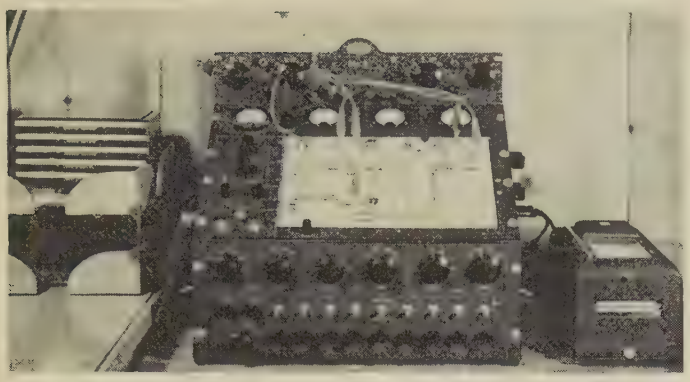


Fig. 1—A general view of Deac'n.

and without the necessity of operating them on a high duty cycle. For the very complex problems, a grouping of these computers will provide a large problem solving capacity at a fraction of the cost of the large computer installation.

Description of Computer

A photograph of the Deac'n is shown in Fig. 1.

The computer itself consists basically of nine high-gain dc amplifiers, mounted in three rows of three shown in Fig. 2. The inputs and outputs of the amplifiers are connected directly to female banana-plug receptacles. The fiber glass plug boards, on which the problems are set up, plug directly into these receptacles, and when in use the board is flush with the top of the computer.

The output of any amplifier can be monitored on one of the three voltmeters provided for this purpose. The time history of any two voltages can be recorded on a two-channel oscillograph, special amplifiers for which are provided in the computer. Selector switches are provided for choosing the functions to be monitored and recorded, and for adjusting the meter and recorder sensitivities. On-off switches and centering adjustments are provided for the individual amplifiers.

The power supply, which forms the base of the computer, supplies regulated plus and minus 300 volts and unregulated 6.3 volts for all of the amplifiers. Its power drain is 250 watts from a 110 volt 60 cycle source.

Some additional features which are convenient to the solution of problems containing nonlinear and time varying parameters are provided. These include two low impedance plus and minus bias voltage sources which can be switched in steps of 1.5 volts. Also provided is a precision potentiometer driven at constant speed by a timing motor.

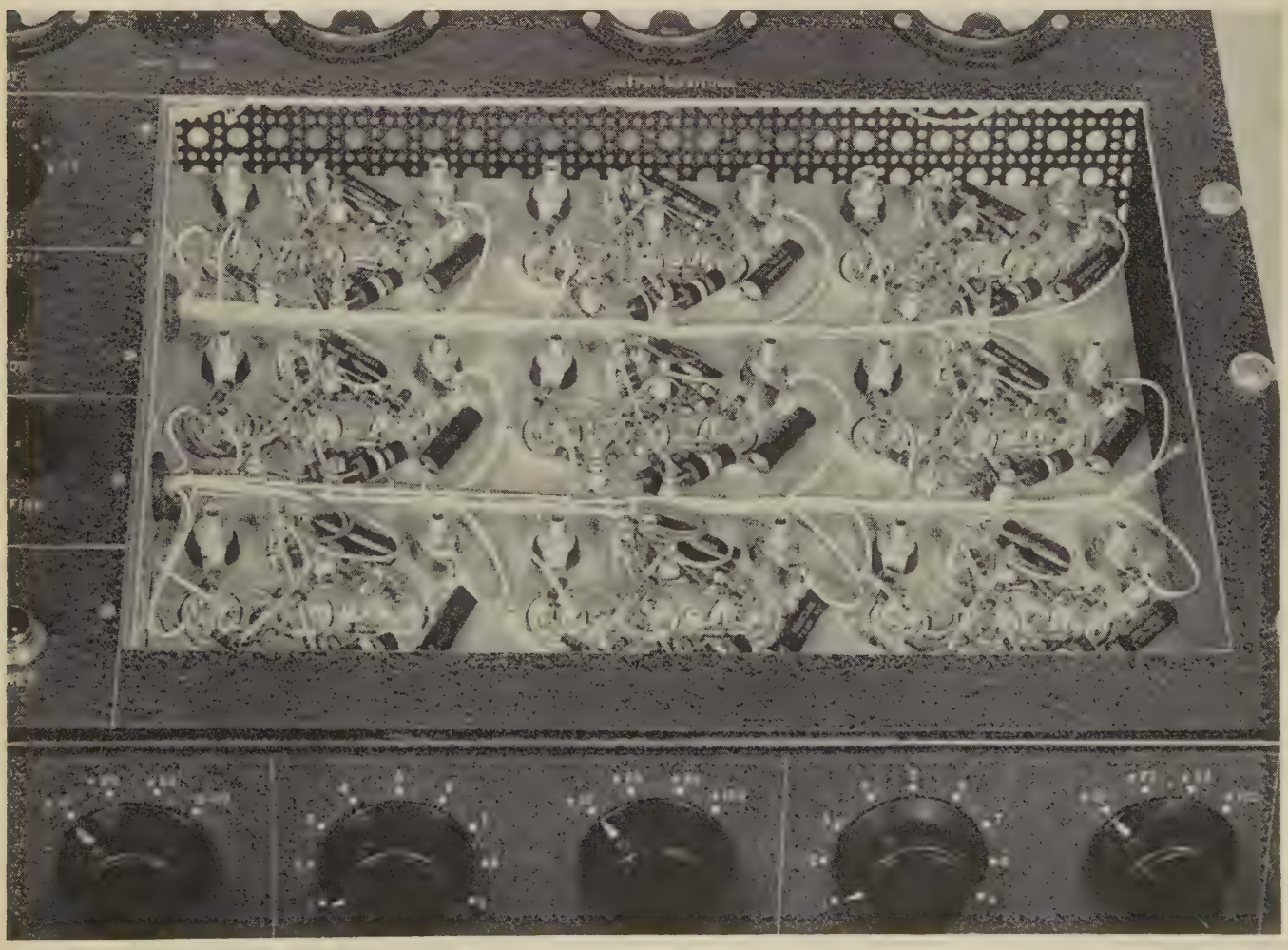


Fig. 2—A view of the amplifier panel in the computer.

Design of Amplifiers

The considerations involved in the choice of the basic computing amplifier are the following: gain, drift, frequency response, output voltage range, compactness, and power requirements. The balanced differential amplifier of Fig. 3 was chosen as the basic amplifier. An analysis of a differential amplifier has been given by Goldberg.[1] The open loop gain of the amplifier shown

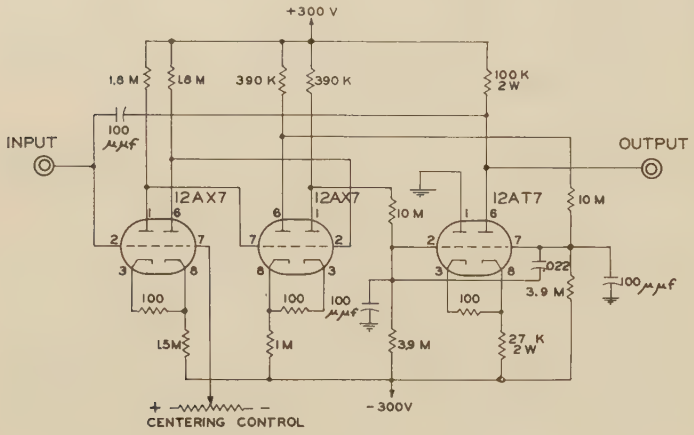


Fig. 3—Computing amplifier.

in Fig. 3 is over 20,000 which is high enough for the types of feedback networks normally employed so that the closed loop amplifier response is accurately determined by the feedback to input impedance ratio. The drift rate is less than 1 volt per hour when used as a 1 second time constant integrator, so that chopper stabilization was deemed unnecessary. The frequency response is described by the fundamental time lag of the amplifier, which is 0.0001 seconds for a megohm feedback resistor and is proportional to the feedback

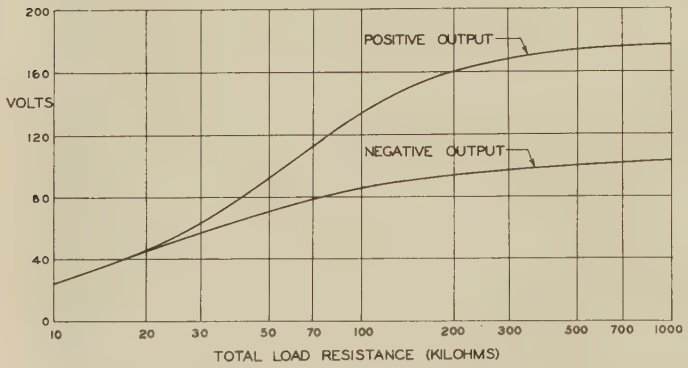


Fig. 4—Saturation level of computing amplifiers.

resistance.[2] The output voltage range is given as a function of load resistance in Fig. 4. The three miniature double triodes and all associated components of an amplifier occupy an area of $2\frac{1}{2}$ by $3\frac{1}{2}$ inches, so that all nine amplifiers fit easily on the 9- by 12- inch plate whose size determines the plug board dimensions. Low power requirements of the amplifier are achieved by the use of high impedance circuits in the amplifier itself and in the external networks.

The above considerations are common to the amplifier design of all electronic analog computers. In the Deac'n there is the additional requirement that the amplifiers remain free of parasitic oscillations for various feedback impedance configurations. Achieving this stability while maintaining the desired frequency response was one of the major design problems of the computer.

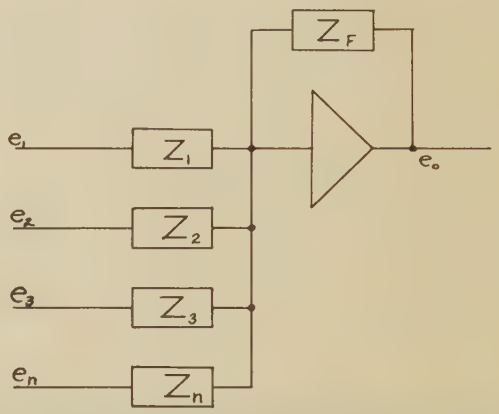


Fig. 5—General feedback amplifier.

Operation of Computer

The method of solving linear problems on this computer is fundamentally the same as that used in all electronic analog computers.[3] However, the techniques do differ in some respects, as discussed below.

The Deac'n amplifier may be used as a flat amplifier with any gain up to 100, as an integrator with an integration time constant from 0.01 to 10 seconds, or as a more general transfer function. The general feedback amplifier is shown schematically in Fig. 5. For very high open loop gain, the voltage output of this amplifier is given by

$$-e_0 = \frac{Z_f}{Z_1}e_1 + \frac{Z_f}{Z_2}e_2 + \frac{Z_f}{Z_3}e_3 + \cdots + \frac{Z_f}{Z_n}e_n$$

where $e_0 \cdots e_n$ and $Z_1 \cdots Z_n$ correspond to the voltages and impedances indicated on Fig. 5. In the useful frequency range of the computer (0 to roughly 100 cps), the use of inductance in forming these impedances is impractical, so that the impedances are necessarily resistance-capacitance networks. It is apparent that with the resistance and capacitance levels completely at the command of the operator, the frequency response and dc-gain of an amplifier may be chosen independent of each other.

[1] L. Goldberg, "Universal direct-coupled amplifier," *Electronics*, vol. 24, pp. 128–131; October, 1951.

[2] For example, G. A. Korn and T. M. Korn, "Electronic Analog Computers," McGraw-Hill Book Company, Inc., New York, N. Y.; 1952.

[3] C. A. Meneley, and C. D. Morrill, "Application of electronic differential analyzers to engineering problems," Proc. I.R.E., vol. 41, pp. 1487–1496; October, 1953.

In addition to the transfer functions obtained by using amplifiers, the problem-solving capacity of the computer is greatly increased by the use of passive networks not directly involving amplification, for which ample space is provided on the plug boards. In this manner, many of the shaping networks used in the system being simulated may be directly duplicated. The use of such networks in the feedback circuit of an amplifier may yield additional functions which would otherwise require a larger number of amplifiers. A simple example of this type in which a three-pole Butterworth filter is achieved with one amplifier is shown in Fig. 6.

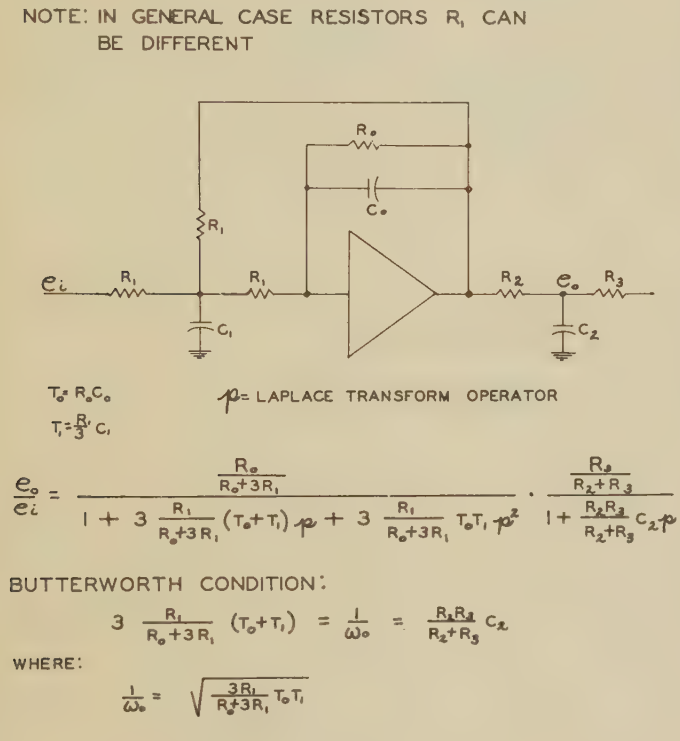


Fig. 6—Three-pole Butterworth filter.

Problems involving nonlinear parameters in which the nonlinearities can be represented by straight line segments are simulated by the use of biased crystal diodes, which are wired onto the plug boards in the same manner as the linear components.

For the types of problems normally solved on the Deac'n, it is usually sufficient to examine the response to a specified set of initial conditions. For this purpose, a variable voltage source is provided for producing step functions and setting integrators to the desired output levels, and two switches for starting the problem. This may take the form of applying voltages or closing connections between amplifiers in the analog.

The components used in constructing the amplifier networks are soldered directly to the amplifier terminals, reducing the hum pickup. The precision of these components determines the accuracy of the analog. The components are usually 5 per cent half-watt carbon resistors and paper-dielectric capacitors. The effects of the tolerance levels will, of course, vary with the application. For example, the corner frequency of a noise filter using a resistor and capacitor will hardly be of sufficient importance to require precision components. However, the loop gain of a feedback system involving a considerable number of amplifiers would certainly require a more precise determination than the product of the nominal gains of the individual amplifiers. In such instances it is customary to measure the gain—either over-all or amplifier by amplifier, as the occasion demands—after the analog has been constructed. The final gain may then be adjusted to the desired accuracy by trimming a single resistor.

The use of fixed resistors throughout the analog has eliminated the wire-wound potentiometers associated with most electronic analog computers. Aside from their expense, the use of such potentiometers has the additional disadvantage of increasing the power requirement of the amplifiers because of their comparatively low resistance. In addition, incorrect potentiometer settings and circuit connections often provide a source of error. The presence of the completely wired analog in the form of a schematic on the Deac'n plug board represents one of the greatest advantages possessed by the computer. The infrequency of errors in amplifier interconnections or component values occurring in its eight months of operation testifies to the efficacy of this method, and has resulted in a most encouraging confidence in results.

Although the problem-solving capacity of the Deac'n is, in most applications, comparable to that of a conventional analog computer, the use of only nine amplifiers has drawn some criticism. The fundamental question appears to be the extent to which functions of secondary importance are simulated in a given problem. It is the feeling of the authors that it is desirable to represent complex secondary systems by simple equivalents, having the correct frequency response in the important frequency range of the main system. This method of representation not only greatly reduces the complexity of the analog, but enables correct solutions to be obtained more rapidly.

The Deac'n was conceived as a tool of the individual engineer rather than of the organization, and to this end simplicity in operation has always been stressed in order to insure one man operation. The ultimate object has been to give the design engineer the freedom to experiment with new ideas as they arise without regard to schedules or cost.

In addition to the standard applications of an electronic analog computer, the Deac'n has been found to be a useful laboratory tool. Because of its portability, it has been used in conjunction with laboratory tests of missile components, servo systems, and subassemblies. The main noncomputing application to date has been in recording of responses of components under test, when such recording has required the use of special filter circuits which were constructed on the plug boards.

This computer should also be useful in engineering schools, both for the solution of problems and the teaching, in related fields of analog computing methods.

Conclusion

A description has been given of an analog computer which is unique with respect to size, cost, and manner of operation.[4] It is smaller than any analog computer known to the authors. Aside from many additional uses to which the computer can be put because of its easy portability, it introduces a new concept of analog computer application. This concept is the widespread use of analog computers in the way that arithmetic desk calculators are used today. This is also made possible by the low cost of the computer, which is determined, in the final analysis, by the size and complexity of the device.

Thus, every effort has been made to keep the design as simple as possible. However, this was not allowed to interfere with the proficiency of the computer in performing its primary task of computing. Indeed, this task has been made more pleasant by the method of problem layout, which reduces the operator's concern over whether or not the problem is set up properly. The problem solving capacity of the computer has been effectively doubled by the utilization of passive networks in the analogs. This, combined with the complete freedom in the choice of amplifier characteristics, has resulted in a capacity which compares favorably with that of commercial analog computers.

[4] Desk type electronic analogue computers are available from the Reeves Instrument Corp., New York, N. Y., and the Davies Laboratories, Riverdale, Md.

Acknowledgment

The authors wish to acknowledge the contributions made by J. W. Miller, formerly of Raytheon, and A. V. Kozloff to the design and construction of the Deac'n.

1954 IRE National Convention Electronic Computer Session III

Foreword

The third Electronic Computer Session at the 1954 IRE National Convention was an experimental discussion on how to make computers more autonomous and how to make them self-repairing. The following pages are a transcript of that session, edited to make it more suitable for print. Each speaker's remarks have been edited by the speaker, which is why it has taken so long for this publication to appear. Some speakers merely corrected their grammar and some went considerably further, but the gist of the original discussion is still there.

The operation of the session is described in the transcript. The participants were asked for their comments on the session, and the most frequent comment was that there should have been fewer original speakers with more time allotted to each one. Comments on the topics chosen ranged from a request for more immediately practical ones to a request for even more "science fictional" topics. The question was raised as to whether the proceedings deserve the respectability of print, and I would appreciate any of the readers' comments, in care of the editor, on this subject. To me, the session accomplished its major purpose because, when the formal proceedings terminated, the participants and some of the audience members gathered in several informal discussion groups and continued to talk for at least another half-hour.

At the conclusion of the session a discussion on germanium diode testing was held in round table fashion at the other end of the room. A summary of this discussion by the moderator is included here.

—*Morton A. Astrahan*

The Electronic Computers III Discussion, Session 51, of the 1954 National Convention of the IRE, held on Thursday afternoon, March 25, 1954, in Morse Hall at the Kingsbridge Armory, Bronx, New York, convened at two forty-five o'clock, Mr. G. M. Amdahl, of International Business Machines Corp., Poughkeepsie, N. Y., presiding.

Mr. Morton Astrahan: I would like to welcome you all to this discussion session of the Electronic Computers Group. If it seems to you, any of you, that the order of business here is being made up a little on the spur of the moment, why, you are very astute.

This session has been organized very strictly as an experiment, on the basis of "nothing ventured, nothing gained." We wanted to see what could be done in the way of doing something experimental in promoting information transfer between people and getting the audience to participate a little bit in what was going on, and we hope to learn something from how things go today.

I would like to introduce you to Dr G. M. Amdahl, who will be the moderator, and will explain to you how this session will operate—I hope!

Chairman Amdahl: Thank you, Mort. A microphone is provided at each table and one for the audience. Anyone who wishes to speak at any table, please use the microphone. I have control over these microphones, and if you wish to have your voice come out of the loud-speaker, please attempt to attract my attention first.

Any talk, including the first talk by the discussion leaders, will be limited to five minutes. If you wish to talk longer than that, at least let one other speaker intervene.

We will, first of all, call on each of the discussion leaders in turn, and after each of them has had his opportunity to give his talk, then the general audience participation is invited. The speakers will be essentially the moderators for the people at their table, and it is expected that they will manage to keep the discussion limited somewhat, unless they actually are connected to the p.a. system at that time.

Our first speaker for today, who will give a five-minute talk on how to make computing mechanisms self-repairing, will be Dr. Claude Shannon, of the Bell Telephone Laboratories, Inc., Murray Hill, N. J. Dr. Shannon, Table 7!

Dr. Claude Shannon: Starting off the talks, I can only say a few generalities and platitudes about the general problem. I certainly cannot tell you how to make machines self-repairing, but will speak around the subject a little bit.

Once a large scale computer has been built and debugged sufficiently, there are still failures in operation, errors that the machine will make, due to the failure of the components, and this will continue as long as the machine is in operation. The failures of components are of two general types. The first are more or less temporary failures, such as, for example, the failure of a relay contact due to a particle of dust. These failures frequently will repair themselves. If the relay is used again at a later time, the dust particle may be gone and no significant damage is done to the components.

The other type of failure is failure of a permanent degenerative sort, the cathode emission of a vacuum tube declines, or a re-

sistor or condenser burns out. In a case like this, it is not to be expected that this condition will repair itself, and the only thing to be done is to replace the component or to build the components in such a way that such failures do not occur.

There are two general ways to try to improve performance. One is to try to build better components, to improve their reliability, their life, and so forth. Of course, such research is going on all the time. But, eventually, we get to a point where it seems impossible to further improve the reliability of components. Relays are so reliable nowadays that their errors occur perhaps once in 10^7 trials.

However large, computers have so many different components, the problems which they solve are so long, the elements are operating so rapidly, that even this reliability is insufficient. These factors make it necessary to improve the circuits so that even though the components themselves fail, the circuits as a whole will keep going.

This is done in various ways in some of the machines that are already built. In the relay computers, for example, relays nowadays are almost always built with bifurcated or parallel contacts. A bit of dust in one of them will not ordinarily prevent the other from operating.

Another type of local check has been used in relay circuits. It is required, for example, that in a group of relays representing a number, a certain number of these relays must be operated in order to represent a proper number. If the number operated is less than this particular number, or greater than this particular number, the check is not satisfied and the machine will stop. This is a rather nice type of check, because it not only avoids mistakes, but the machine has stopped at the point where it was about to create an error, and a maintenance man can locate the cause of the error rather readily.

When this type of computer is run at night, it can be set up in such a way that if it comes to a point where the check does not come through, the machine can back up a few steps and try again. In general, on the second try the machine can get past this point and go on. This is due to the intermittent type of errors, dust in contacts, and so on.

The automatic telephone system is really a very large scale computer which operates without significant or really serious errors over periods of many years. The reason why the problem there is so much easier than in the case of an arithmetic computer is that, in general, the telephone exchange does make a large number of small errors, but they are not very serious. That is, relays are failing, calls are failing to go through on the first try, and the machine tries again, and so on. None of these small failures disable the whole machine in any real sense of the word.

Another case where a large amount of error-correcting has apparently been built into a machine is the case of the brain of animals or man. There you have a machine which is far larger in numbers of components than digital computers, and it runs a much longer time without errors. Furthermore, there is no maintenance, in the sense that we go in and take old components out and put new components in. I understand that the neurons are never regenerated in the human brain; nevertheless, the brain apparently operates without serious errors for great lengths of time. Seldom do you see errors of the type so common in digital computers, with a stream of nonsense coming out of it.

Chairman Amdahl: Thank you, Dr. Shannon. Our next speaker is Mr. Nathaniel Rochester, of the IBM Engineering Laboratory, Poughkeepsie, N. Y. He will speak on making computing machines self-repairing.

Mr. Nathaniel Rochester: I am going to start by stating a figure that I am not going to defend very strongly, and that is that the complexity of our most complicated machines seems to increase by a factor of about 100 every fifteen years. The way I figured this out was to consider that if in 1938 someone built an electronic device with 100 electronic switching elements, he would be considered rather bold. Then, in 1953, fifteen years later, if we built them with, say, 10^4 switching elements, that was quite a few, but not unreasonable.

Now, if we extrapolate thirty years, we find that in 1983 we would have machines with 10^8 switching elements.

To see what that means, one thing we can do is compare it with the brain. The brain contains 10^{10} neurons, and each neuron is perhaps 1,000 times more complex than a switching element, so that the brain is still 10^5 times more complicated than this device we are talking about.

Another thing to be considered is reliability. If we have a device with 10^8 switching elements, how often is one of them going to go out of commission. By a similar, but even rougher process of figuring it, the reliability of our components seems to be increasing by a factor of ten in every fifteen years, and if that is right—and this is very, very rough—the average life of a component, electronic switching element, in 1983, would be two million hours. What that means is that every 72 seconds one of them would fail, on the average.

This average could be considerably improved if we extend the techniques that we are using for marginal checking and getting rid of components that are going to fail, all at once, before operating. Nevertheless, there is a real problem of maintenance. It is going to have to be faster than it is today if we are going to succeed in making machines like this out of components as poor as this.

We might consider that just a few years ago the way a machine was maintained was that the maintenance man would try and find the trouble, and after he found the trouble he would replace the part. Now, with our larger machines, the maintenance man uses the machine itself to a considerable extent to find the parts that are defective, using diagnostic programs.

This is still rather slow, because he has to decide what to try, and it isn't anywhere near as fast as it could be. With these more complicated machines, the machines in the near future, it may be possible to reverse this process a little bit and let the machine find its own trouble and use the maintenance man as a device to plug in a new component.

In other words, with much better diagnostic programs, it should be possible for a machine to decide to replace sub-assembly 198, and then the maintenance man would quickly obey and replace the component.

Then, of course—and it is hard to visualize it—the ultimate stage is where the machine replaces the parts itself. A machine that is not working isn't going to succeed very well in replacing parts, and so something is going to have to be working still if the part is going to be successfully replaced. One possibility is to have two machines which would repair each other. This is really quite reasonable.

Another possibility is to have machines in which a basic element, assumed to be always running, and repairs less essential parts around the edge.

What these possibilities seem to imply is that eventually the maintenance people will be relieved of routine and simple chores and will be left with only the unexpected or excessively complicated cases.

Chairman Amdahl: Thank you. And now, at Table 5, we have Dr. H. P. Huskey, from the Institute for Numerical Analysis, UCLA, West Los Angeles, California. He will speak on both topics, making computing machines self-repairing, and making computing machines more autonomous.

Dr. H. D. Huskey: The first point I would like to make is that the language which we use in "speaking" to these computing machines is rather elementary and undeveloped. In fact, you could think about it as being similar to what might be called the "finger language" stage. The process of counting on fingers must have been the first language used in the science of mathematics.

Certainly great advances were made when Arabic numerals and other efficient notational schemes were introduced. In the development of mathematics a great many of the accomplishments have been possible only on account of an efficient method of notation, and this is one of the things that we need in using computing machines.

Indeed, we need to get away from the situation of spelling out in detail the additions, the subtractions, the multiplications, and so on, that the computer must perform to carry out a particular computation. This is being done to an extent. Most groups that are operating computing machines make use of subroutines to accomplish more complicated arithmetical processes.

For instance, some groups are using interpretive coding systems in which use is made of a command system that is entirely different from that designed in the computer. There has been research going on to increase the possibilities in this direction, and one of the goals for which I am striving is to be able to approximate the situation in that what is prepared for insertion into the computer is essentially the information that is typed on the manuscript that represents the problem. In other words, what is desired is for it to only be necessary to type the formula that is to be solved, or used in the solution of the problem, into the computer, as versus interpreting this formula for the computer as a list of subroutines or commands that it must obey.

In considering the design of computing machines, more and more thinking is being done in terms of building blocks consisting of certain logical units, as evidenced by a

number of the papers presented at this session. This will make it possible to design a computing machine by a list of logical equations, however quite a complicated list is necessary. The list, in fact, is so complicated that it is worthwhile to devise methods of simplifying the set of equations that represents a computer. Some work has been done in this direction in the simplification of certain types of logical truth equations. There is a need for more of this type of development, and it is certain to take place. When methods are developed and perfected, this will make it possible, in my estimation, for a computing machine to be used to improve the design of the next computing machine. You can perhaps think of it as a bootstrap process.

As you walked into this room you may have seen a sign on the door about modular production of electronics, which is a mechanized production system for electronic units. This is not completely nonhuman at the moment, but certainly very nearly so. If one of these systems would be connected to the end of the new machine that the computing machine is designing, you can imagine that you might have a reproductive process which would make it unnecessary to repair these machines. Instead, as was mentioned by someone today, when a human being finally gives up, you do not try to repair him, you simply bury him. Thank you.

Chairman Amdahl: Thank you, Dr. Huskey. Our next speaker, at Table 6, is Mr. John W. Mauchly of the Remington Rand Company, Eckert-Mauchly Division, Philadelphia, Pa. He will speak on making computing machines more autonomous.

Dr. John W. Mauchly: One of the people told me at lunch that he didn't think it was possible to say anything in five minutes. I think his five-minute contribution proves that you can, and I will try to do as well in my five minutes.

It seems to me that I should talk about education. This meeting is being held as an experiment in a kind of education, and concerning computers a great deal of education needs to be done, because there are still many people who should be using computers who are not benefiting at all from them. One of the things which people need to know more about and on which more work should be done is in making computers more autonomous.

We can hardly expect that a great deal of progress will be made in this field until lots of people realize what can be done in this direction, and that there is a profit to be made by so doing. If enough people realize this, then adequate work will be done on this subject.

There are various directions, of course, which such development may take.

What do we mean by making computers more autonomous? The various people here today are free to interpret this as they see fit. One interpretation is as follows: Is it possible to have the computer do more toward organizing its own programs and routines for carrying out the operations which we want to have done? Under this interpretation, we shall be the master minds who direct what the computer shall do, but we don't want to do all the drudge work ourselves, since the computer itself can be made to do such tasks if we do more of the masterminding first.

This leads us into the subject of automatic coding and programming. This is much more than merely the translation of symbols which are convenient to human beings into other symbols convenient to a machine, although such translation is one part of the entire job.

There is a great deal of analysis needed on convenient language and organization, so that the various people, all of whom are interested in similar problems, can work together on this group problem. In the *designing* of computers there is an advantage in building many computers to the same design so as to cut down the cost of production. Similarly, there is an advantage in *programming* to have many people able to use the same program, so that it doesn't have to be thought out over and over again.

At present, for every large computer, there are programs for finding the characteristic roots and vectors of a matrix—to give but one example. Such programs really need to be thought out only once. Actually, however, such routines have been independently programmed for each computer, and the same thinking has had to be done many times.

Now, if the time hasn't run out on me, I want to comment on the comparison between the computer and other entities which supposedly can think, such as the human brain—which does think occasionally. Sometimes people say that although an electronic computer has very few elements, or neurons, compared to the human brain, it nevertheless has a big advantage because it is so much faster. We might turn this around and say instead that, since the electronic computer is so much faster, we may therefore make it mimic some of the simpler operations which we now call thought.

I am not going to attempt to define that. It has been remarked that anything you think is thinking, *and are able to define*, can probably be programmed for a computer. The complexity of the computers that are available today is perhaps more than you have personally realized. We frequently evaluate computers in terms of their *internal* storage capacity, and are prone to rate them in terms of the speed with which they multiply or do other arithmetic operations. However, in using a computer as a model for some other system, it is not necessary that the model perform in real time. If it is not too costly, it is enough that the problem be solved even at a very slow rate, so long as the model is a reasonable one and bears a recognizable similarity to the system of interest.

So, in closing, I want to point out that one computer with which I have been associated has under its automatic control 10^8 bits of information. No human hand needs to intervene during its operation on this quantity of information. It can play away with those 10^8 bits entirely automatically. As was brought out by Turing when he wrote his paper on computable numbers, each of these bits of information can be reeled into the machine and govern the action taken by the computer. The action taken at each step is recorded in terms of other bits of information, which can affect subsequent steps. Hence, present equipment can be made to perform as a sort of Turing machine using 10^8 bits. In conclusion, then, there is much that we aren't doing now which we could do with present equipment to advance the study of computing machines in accordance with such a general program.

Chairman Amdahl: Thank you, Mr. Mauchly.

The next speaker is at Table 8, Mr. E. F. Moore, of Bell Telephone Laboratories, Murray Hill, New Jersey. He will speak on making computing machines self-repairing.

Dr. E. F. Moore: I would like to peer a little farther into this crystal ball Mr. Rochester was looking into, particularly in the last stage, where the computers themselves can replace the faulty components. What are some of the logical problems, the organizational problems, that will have to be solved before this can be done?

We might imagine as the goal a case where we make a computer out of a large number of components that fail at random, and individually have a certain life expectancy, but that we organize them in such a way that the life expectancy of the machine as a whole is greater than any of its components.

Now, how can we do this? For instance, if defective parts are to be removed, that is, if the machine has a soldering iron to unsolder the component that goes bad and put in a new one, what happens if something goes bad with the soldering iron? Or, more tricky than that, what happens if something goes wrong with the circuit that controls the soldering iron, or the circuit that decides when something has gone bad?

Each of these difficulties may be looked into, and an over-all organization thought out. For instance, the example that was proposed, that of having two machines repair each other—that would undoubtedly be safe against some kinds of errors, some kinds of failures of the components; but in the case where one of the machines has such a failure that causes it to think that something is wrong with the other machine, then each of them will start unsoldering the other one, and so there will be need of a referee, and then, we have the problem of what happens if something goes wrong with the referee.

If we try to make this completely safe, as far as we can go—what the ultimate is that we can achieve in this direction, certainly has not been reached. All of these straightforward methods of arriving at it, taking two machines or taking three machines, or taking one machine to repair another machine, do seem to have a loophole in them; but by looking for these loopholes and by looking for the sort of reasoning by which you arrive at them, it should be possible to organize machines in a little bit more subtle fashion to get around them.

I feel confident that this can be done. As to how soon anybody will actually carry out any detailed mathematical proofs showing that it is really possible, that is another question. Then how soon it will occur that somebody actually builds this machine, that is a horse of another color. They may run into a lot of practical difficulties, particularly if the kinds of failures that the parts can actually have are different from the

kinds of failures that were assumed at the outset.

What happens if something else goes wrong that wasn't foreseen? So, to have this ultimate flexibility, to have a life expectancy greater than its components, you probably should have built into such a machine a certain amount of autonomy for it to do things on its own, to learn as a result of the experience it meets with, and so on.

The two topics that we have been talking about are thus not completely unrelated to each other.

But how long it will be before machines that go this far will be actually attained is something that somebody will have to use a little bit better crystal ball on than I am using.

Chairman Amdahl: Thank you, Mr. Moore.

Our next speaker, sitting at Table 2, is Dr. L. N. Ridenour, of the International Telemeter Corp., Los Angeles, Calif. When we asked Dr. Ridenour what he was going to talk about, he said, "Making computing machines more autonomous includes both subjects. Therefore, I will choose it."

Dr. L. N. Ridenour: Well, that saves me fifteen seconds of my time.

To investigate this question of autonomy one must inquire what people do in behalf of computers; there are four activities that fall in this category. People design and build computers, people program problems for computers, they fix them when they break, and there is currently a vast amount of routine and mostly wasted human effort employed in translating material from human language into language understandable by the machine.

Now, I shall assume that we can usefully exclude from the present discussion the problem of building a machine that will design and make machines, not because it isn't germane or worth talking about, but because the problem of doing it is reasonably straightforward. I think that this could be done on the basis of known and existing techniques.

I shall make two remarks that may have some significance in the present connection, in terms of lowering the work of programming, machine repair, and routine translation for input and output.

To take the simpler one first, the development of an adequate reading machine which will translate, into terms understandable by a machine, the kind of writing and printing that is intelligible to human beings, would reduce substantially the vast amount of routine keypunching that now goes on.

A machine to do this has to be a rather sophisticated one, because we do a rather sophisticated job in communication techniques. In particular, we take context into account to a very high degree, and thereby get over difficulties of unintelligibility that would otherwise be insurmountable.

Here, again, while this is a difficult task, it is not necessarily one that involves techniques which are today uncommon.

There are two other things (which people piously talk about but nobody does anything about) that may be of significance in connection with our present topic, and I should like to mention them.

People have compared large information-processing machines with the nervous systems of animals; there do seem to be similarities. There are two glaring differences which seem of great significance. The first is that, in an animal's nervous system, reliability is achieved by redundancy. You never depend uniquely on any single element of the system to do anything.

Also, there is the problem that has been referred to by some of the other speakers: that of deciding whether a disagreement between two redundant elements should be resolved in favor of one or another. That problem is avoided entirely in nature, because nature uses sufficiently many redundant elements so that the one which is out of step is immediately apparent. If you have 100 channels for a message, and there is a 1 per cent likelihood that one of them can go sour, then all you need do is believe the 99 neurons that agree. This principle is made use of in nervous systems, and I think it should be made use of in any complex machines that must be highly reliable.

Finally, a natural system does have the advantage, as Dr. Huskey has said, that when it fails you can take it out and bury it.

Now, this animal nervous system that has been in existence for 50 millions of years is very good and very reliable. Each of us has a very fancy machine in his head which operates for three score and ten years and uses only 25 watts. If we look at what we know about the design of this marvelous machine each of us has between his ears, the second principle which emerges is that very great use is made of the fact that the mind has the property of retentive learning.

Now, this is a way in which we handicap machines very, very severely. We cause them to be born again each time a program is finished and a new one is to be put on. Except for subroutines stored in the library, there is no history or record of what the machine has done in the past. The subroutines in the library aren't available to the machine unless a machine operator makes one available, in the course of a given problem. Our inability to build large enough memories—large enough high-speed storage units for these machines—is the main thing which so handicaps present-day machines and keeps them so far from being autonomous.

Chairman Amdahl: Thank you, Dr. Ridenour.

Now, at Table 1, we have Dr. A. L. Samuel, of IBM, Engineering Laboratory, Poughkeepsie, N. Y. He will speak on making computing machines more autonomous, and on making computing machines self-repairing.

Dr. A. L. Samuel: I find myself in a rather embarrassing position, that of being the seventh man on this panel without even a seventh-inning stretch, and following speakers who have discussed this subject so very thoroughly. Of course, this predicament was not entirely unexpected. I knew I would be seventh on the panel, but I had thought that some of the speakers would say things with which I disagreed violently, and that this disagreement would do in lieu of a set speech.

Unfortunately, or perhaps fortunately, I do agree with most of what has been said. I do have a few remarks that might be added by way of filling in.

First of all, about this problem of making machines autonomous. I think we mean two quite different things by this. One is in the terms of Dr. Ridenour when he said "making them repair themselves was a phase of autonomy." But we also think of them as being autonomous in terms of being able to perform problems with less detailed instructions than they now require. This latter problem has not been discussed.

In the engineer's language, this is simply a problem of matching impedances. Machines are designed at present to speak quite a different sort of language from their users. At the present time, it takes rather a high degree of intelligence to program present-day machines. In fact, it is just about impossible to obtain enough people with the required training and skill. One of the big problems we face is that of designing machines which understand something more nearly like ordinary human language.

Now, as to this problem of reliability, in terms of making machines repair themselves, I would like to point out an analogy that I think is important. Most of you, I assume, know something about the problem of designing error-correcting codes. A previous speaker, Dr. Shannon, pioneered in this field. He pointed out that by introducing redundancy into a code conveying intelligence, one can make the code independent of certain types of errors. You can, if you wish, add a limited amount of redundancy and so detect single bit errors, or you can add more redundancy and correct simple errors or even detect multiple types of errors.

Anyone who has studied this subject knows that the amount of added information required in terms of the number of bits you are using to convey the message—that this fraction goes down very rapidly as the number of bits of information in the original message goes up; and, specifically, if you have two to some power "n" bits of information, you have only to add something in the order of "n" bits of information to correct single errors.

Now, I submit the proposition that the problem of building complex machines is somewhat analogous. The kind of trouble that occurs when single elements fail is quite analogous to errors in codes and, by analogy rules that have been developed for the coding problem should be applicable to the problem of designing complex machines. We have not done this, although I submit that this is an interesting approach. Where it may lead to, nobody knows.

There is one caution that should be mentioned. Consider an error-correcting code which will correct single errors. If a triple error occurs the device will interpret this as a single error and will attempt to correct this apparent single error. Usually some bit of information that was right will be suspected of being wrong, and will be altered. Therefore, you have the phenomenon that any device that tends to correct a simple type of error will, in the presence of a more complicated error, usually make matters worse.

This is quite analogous to the case of the two machines that are to repair each other, when one of them erroneously decides that the other is incorrect. We must face this fundamental problem: That as we make

more complex machines in an effort to correct certain types of errors, we will inevitably make the device subject to more serious difficulties in the case of complex errors. Thank you.

Chairman Amdahl: Thank you, Dr. Samuel.

And now, at Table 4, we have Professor J. B. Wiesner, from the Department of Electrical Engineers, MIT, Cambridge, Mass. I asked Professor Wiesner what he would talk on today, and he just said that he would speak.

Professor J. B. Wiesner: I happen to have the distinction of being the only member of this panel who has not had some intimate relationship with a computer. I really was not certain why I was asked to participate in this program. On the other hand, my ego didn't permit me to turn down the opportunity to appear with Drs. Shannan, Samuel, and Ridenour who are here, and Dr. von Neumann, who was unable to come. I had one bit of wisdom which I intended to give to you, but unfortunately I mentioned it to Dr. Samuel during luncheon. So, you see, I am in a position of having very little to say on this subject.

I suppose, as the last man on this team, I ought to be mad at computers. I understand our positions were selected at random by one of the IBM machines, and it made me tail man on this program.

I should try to sum up the positions of the previous speakers on these two questions, and it is easy to do. The answer to both seems to be "Yes."

By the way, I know most of these individuals—I have seen their machines and I have seen them try to make them work, so I understand why they want to turn the responsibility for the repair over to the machine.

Seriously, I think there is one very important point to be made here, and it has already been said implicitly, if not explicitly. The point is that the extremely difficult part of this problem is the diagnostic job. Maybe repairing the equipment is also difficult, but certainly the important thing is to be able to recognize in some more or less foolproof way what is wrong and what corrective action should be taken. This is the most difficult problem that faces the person who must repair a machine or the machine which is to repair a machine.

In computers, as in other electronic gear, preventive maintenance is very important. The old practice of going to the dentist once every six months or seeing your doctor regularly is probably a good idea for machines, too, and in some machines this is what the designer has tried to do. You go around and tap the equipment on the knee to see how it is feeling. If a particular tube has drooped, or if some voltages are wrong, you take corrective measures before the machine has failed seriously.

At MIT there have been a number of attempts to make electronic machines, other than computers, correct their faults, and I managed to corral a man, who is sitting beside me, who has attempted to organize certain digital data processing equipment so that when a subassembly failure has taken place, or when some component is in danger of failing, either subassemblies or full units are replaced. Obviously, when units are very large, you don't care to switch complete machines.

I think the point about the parallel channel multiple circuits made by Dr. Ridenour and the question of using redundancy in communication systems or computing systems to correct errors are really different ways of saying the same thing. We sometimes get frightened by the idea of using a computer with 10^4 vacuum tubes—that seems to be a standard number now—and we don't at all like the idea of having 10^5 elements instead in order to get more reliability. Incidentally, I think that a factor of ten would give you a great deal of additional reliability. On the other hand, if you talk of a machine having 10^8 elements, I don't think it frightens you to make it 10^9.

Mr. Astrahan: Thank you, Professor Wiesner.

I want to thank all of these speakers for their unusual discussions. We will operate from now on, for a while, until a better way turns up, by recognizing each table for general discussion. If the discussion leader or any of the other people at a table wishes to make a comment, will someone at the table raise their hand, and we will recognize the table and turn on the microphone.

Mr. R. M. Fano: I would like to comment on the possible pertinence to the subject under discussion of a fundamental theorem on coding presented by Dr. Shannon in 1948.

The theorem states, roughly speaking, the following. Consider a communication system through which symbols can be transmitted at a rate of say n/sec., but which is disturbed by noise so that the symbols may be received incorrectly. It is possible, by sufficiently involved coding, to make this channel equivalent to an essentially noiseless channel through which symbols are transmitted at a rate of (n/k)/sec. The constant k is larger than one and related to the capacity of the noisy channel as defined in information theory. I should add in this connection that a student of mine, Mr. A. Feinstein, has recently extended Shannon's theorem to the effect that the probability of incorrect reception of a long message vanishes sufficiently fast with the length of the message to yield reliability in an absolute sense rather than just percentagewise.

A computing machine may perhaps be considered as a communication channel in the sense that the input data correspond to the transmitted message, and the output results to the received message. The number of elementary operations per second may correspond to the rate of transmission of symbols in the communication channel. This analogy suggests that it is perhaps possible to program the machine in such a way that it will operate as an essentially errorless machine in spite of internal random errors in the elementary operation, but with an effective number of elementary operations per second reduced by a constant factor.

The perfect reliability of over-all operation implied by the previous statements is approached only as a limit when the total number of elementary operations grows indefinitely.

Roughly speaking this means that as the problem becomes more and more complex, the over-all reliability can get better and better at no further sacrifice of operating efficiency. That is, the ratio of the capacity of the equivalent noiseless machine to that of the noisy machine remains constant.

Of course, internal errors can occur not only because of random malfunctioning but also because of the total failure of physical components. Shannon's theorem seems to imply that the effect of such physical failures can be eliminated in the limit as the complexity of machine itself grows indefinitely together with the complexity of the problem. Again, perfect reliability is approached as a limit while keeping constant the efficiency of operation of the machine.

The point has been made by other people that one cannot talk about a machine repairing another machine because this raises the question of who repairs the repairing machine. In other words, one must consider the machine and the repairing machine as a single system so that the basic question remains that of constructing a system which operates properly in spite of some internal random malfunctioning or complete degeneration of components. Shannon's theorem shows that certain simple types of systems can be made to operate in this manner, at no loss in efficiency, if the systems are sufficiently complex. Of course, the theorem does not prove anything about computing machines but only suggests that a similar type of behavior might be obtainable from them.

Another point that is worth stressing is that the operation of a machine can vary smoothly from perfection to complete inoperability. This amounts to saying that as the machine degenerates internally, the rate at which it can operate reliably decreases continuously to the point at which the machine becomes worthless and is thrown away.

I am very much interested to find out if Dr. Shannon agrees with my speculative interpretation of his work.

Chairman Amdahl: I am going to recognize Table 8 next for Mr. Jay Forrester.

Mr. Jay W. Forrester: I wish to make a few comments on machine repair because I differ with the previous comments on multiplying machine elements to follow the pattern of the nervous system. In terms of the present techniques and the techniques that we are going to have in the next few years, the question of economically achieving the desired result requires some examination. It isn't necessary to build a machine following the system on which the brain presumably operates. We can today build machines which have an initial error rate that is as low as any of the present-day applications require and many orders of magnitude lower than the human brain, even with all its redundancy. Error rate can be low compared to the input or output errors, so I think that an error rate for a machine in proper repair can be achieved that is satisfactory.

Now, the question comes up, what happens when a machine decays and is no longer satisfactory? We have a choice between duplication of elements, or putting sufficient effort into the detection of the deterioration and the replacement of the parts. It seems clear that the best economic solution is to equip the machines with a deterioration detection system which can find the failures before they cause trouble and permit re-

placement of only the faulty devices.

The previous speakers have given the impression that self-repairing is for the future, and that we can only talk about it at the present time. I think it is something that is already partially available to us. A significant step is worth pointing out. A very powerful technique for machine fault detection exists in the combination of the diagnostic program and the marginal checking system. You are all sufficiently familiar with both of these techniques. The diagnostic routine is a way of pinpointing a failure that exists; and marginal checking is a way of converting an intermittent or incipient failure into a continuous failure which the diagnostic routine can find. Combining these two one can make a rather systematic study of the machine and find its weak points. Both have been automatized under the computer's own control giving it the power of self-inspection. Such a system is now operating. A small additional step would be to arrange the machine to adjust its marginal checking parameters for best operation and thereby make temporary compensation for weakened components. This, if done, would provide a form of self-repair.

This combination of the diagnostic program and marginal checking now picks up about 80 per cent of machine failures, and the self-correcting feature that I speak of would lend itself to temporarily treating these 80 per cent. A very large and significant fraction of potential machine errors can now be brought under partial control of the machine.

Chairman Amdahl: Thank you.

The next table to be recognized is Table 3, Dr. H. R. J. Grosch.

Dr. H. R. J. Grosch: I want to speak for that poor, ignored character, the human being. I would like to tell you all about the widely available inexpensive production facilities, the opportunities for worth-while experiment in this region, and so forth, and to say that practically all of the solutions that have been pointed out for problems I am not too sure exist have involved a gush of badly-digested electronics.

I am a consumer of computing equipment, and I would doubt very much at the present stage of technology if it is possible to hang a simple, inexpensive automatic soldering iron on the back of my 701, which will cost me less than the customer engineers who would be eliminated from the IBM payroll.

I am all in favor of you fellows going ahead and working on this thing. I would prefer that you would do it on paper, so that it doesn't turn up in my taxpayers' bill.

However, I think it would be a great mistake for anybody but a few big centers and really bright boys to work on it. The rest of you ought to put out some of the fires that are burning right now.

May I suggest that we ought to carry out this idea of the destruction of the corpse a little bit more completely. It seems to me that we usually only bury the human computers, although I remember the obsequies of a very large monster that used to live about five or six miles south of here. It would be nice if somebody would set up an automatic dismantling project, which we could call "Tinker Destroy" or something, and get back the component parts ready for automatic reassembly into something new.

This should be located in an adjacent area with conveyor belts so that there will be no loss of time.

We must deal with a system that includes the human element. You people are not building computers in a vacuum—or, if you are, I suspect they are selling rather badly. If you are trying to write self-programming programs, in order to work effectively they must speak not the language of an engineer who is building the computer, nor the language of the logical designer who thought up the machine in the first place, nor even the language of the programmer, who is usually a chessplayer and a fairly precise sort of person. It must speak the language of the customer.

I deal with these people daily, and a messier minded bunch you never saw. Possibly poets would be worse, but I doubt it, and until you are able to take their handwriting and turn it into something that a 701 can operate on without much human intervention, I am afraid I can't use it.

Chairman Amdahl: I will now recognize Table 2.

Dr. Ridenour: I sort of lost touch with the continuity, but a while back Dr. Valley confided in me that he knew all the answers to the problem, and I thought it would be nice if he would tell us about them.

Dr. G. E. Valley: Well, thank you, Louis. However, since then, I have heard five or six more speeches and now I realize that I didn't know what the problem was.

It seems to me that the problem of a self-repairing, general-purpose, digital computing machine ought not to be completely divorced from the general problem of repairing heavy electronic equipment in a way, such that the machine which is to be repaired need not be shut down. To whatever extent industrial electronics doesn't all turn out to be composed of general-purpose digital computers, one may rule out from sole consideration methods which depend on error-detecting programs and that sort of thing.

It seems to me that when you set out to have the machines go for a long time, on the basis that they are automatically repaired, you have to decide what the criterion of excellence is going to be and for how long you want them to go on the average, before they die, and how much extra capital investment you are going to make for this privilege. If you don't want them to run steadily for a year, then you oughtn't to pay so much money, and I think this determines how much redundancy in the form of preinstalled spare parts you are going to put in.

You can either build 81 machines in order to get one running all the time, or you can have one machine with 10 per cent spare parts and a program of some sort.

Now, I don't think you need to worry about having one machine making a mistake when repairing another, and so "disrepairing" it. One way of avoiding that trouble would be to have the two sets of parts built together as one machine; with duplicated components and a monitor or comparator component associated with each double component. When the two flip-flops of a pair, for instance, do not agree with each other, you simply replace both of them without asking which is wrong, and the unit of monitor equipment at the same time.

This is a very simple way, and also it allows one to imagine how to mechanize the thing, and I think we cannot avoid this matter. If automatic repair means automatic repair, it doesn't mean automatic calling for a service man. You have got to have a hopper into which you put the flip-flops or Tinker Toys, or protein molecules, or whatever you want to call them, and an automatic way of replacing them, and one way is the way I have just suggested. I presume there are others. Thank you.

Mr. Will Gersch: I am Will Gersch from the AEC Computing Facility at New York University.

Some of the previous speakers have speculated on how computers might repair themselves. There are a few ideas on how computers can be built I would like to air.

Considering the problem of building a computer, I first have a comment on the components, like magnetic devices, that are expected to go into future computers. On that, I would take issue with Professor Forrester.[1] I feel that marginal checking is restricted to things like tube devices, and that newer components will not lend themselves as readily to marginal checking. Therefore, it appears that other means of locating faults are necessary.

What I have in mind for troubleshooting a machine—and that is the first step in self-repair—suggests that first, several distinctions be made in some of the ideas about redundancies of equipment in machines.

There are important differences and consequences in duplication of equipment on the component level, the functional unit level, and the system level. These differences have not been adequately emphasized.

A means of building computers that can be economical in computation time and efficient in using components, would be accomplished by duplication on the functional unit level. The computer would have duplicate units like arithmetic units and busses and counters. It would be organized so as to use the individual function units separately and simultaneously in a rather complicated way most of the time, and then periodically recompose the units so as to provide duplicate checking. In addition to the simultaneous computation of several items, the scheme implies internal autonomy in changing its structure.

I suppose a reference to a biological model is in order. What is suggestive here is the idea of the ability of an organism to modify the functioning of its organs to cope with an impairment.

Of course, this approach makes the assumption that an adequate logic can be constructed. But the construction of a logic is more attractive a problem than the construction of a machine with many components not being used for computation. Thank you.

Chairman Amdahl: Thank you. Due to a shortage of time, I am going to recognize four more tables and allow three minutes only per speaker, after which we will have an intermission. All right, Table 7.

[1] Professor Forrester later indicated that marginal checking is a perfectly general scheme.

Mr. Raymond Lazinski: I am a Research Associate at Jefferson Medical College in Philadelphia. You may wonder why I am here.

Before I took my present position at Jefferson, I was on the Research Staff of the Moore School of Electrical Engineering of the University of Pennsylvania. While I was there, I worked on electronic information handling equipment. My interests in the field of computers has changed its direction from electronic computers to biological computers.

I am glad that the opportunity for me to ask a few questions and make a few comments has risen. It has been stated in the preceding discussion that there is an endeavor to duplicate, in a small way, "thinking function of the brain" by using digital techniques. I believe that the assumption, that this duplication of information handling technique, will be done with the use of standard components, tubes, resistors, transistors, etc., will lead to much misdirected effort in a roundabout approach to the solution of this problem.

The description of the working of the neuron, the building block of the nervous system, has been greatly amplified in the past five years. For a good engineering description of the characteristics of a component, which will meet requirements as a basic building block for a "thinking" information handling system, I would suggest a reading of the journals on physiology, anatomy and neurophysiology of the last years. I would particularly recommend a study of the work of Curtis and Cole, and Hodgkin and Huxley. If you should wish a bibliography on this subject, I suggest the bibliography presented by myself and printed in the rear of the Proceedings of the AIEE, IRE ACM conference of December, 1952, pp. 140–141.

There have been very few experiments performed, to my knowledge, on the problem of duplication of the action of the neuron Some people have tried the iron wire in the acid type of conduction of an impulse; some have made thyratron circuits that trigger other thyratrons in tandem.

The action of the neuron in some cases has been tremendously over simplified by those who state that it acts like an electronic multivibrator. I am sure that a small amount of time spent by those interested in this subject in consulting an elementary textbook on neurophysiology or neuro-anatomy will see that the very vital role of the time and special integration of impulses at the synapse (neuron junctions) is completely ignored by those who site the above mentioned analogy. At the 1952 Pittsburgh meeting of the ACM, Dr. W. S. McCullough presented a paper on the information handling ability of a synapse. Here is a description of a number of neurons acting as information handling components.

I heard some comments a few moments ago about the burying of a biological computer when it began to malfunction, I wish to remind those who concur that in many simple cases a neurological splice can be made and thus in many cases the circuit is restored to operation. It should also be noted that in a number of cases almost one-third of the components of a biological computer have been found to malfunction and, therefore, were removed (a lobotomy or hemispherectomy) with little loss of data within the system. One case in point is cerebral lobotomy, which was performed after which the patient was examined by psychologists and psychiatrists. They could find no loss in information or information handling ability. This would lead one to appreciate the redundancy of information and information handling equipment we all carry around with us.

I wish, in closing, to remind those people interested in developing a self-programing computer that the "neuron" has proven itself. Although the biological neuron is extremely complex, we must remember that it lives, grows, abstracts its energy, etc., by the assimilation of organic materials and gases. These reactions are unknown in most part. I am not suggesting that we try to build a biological neuron. I am suggesting that we use similar techniques and use electric power instead of enzyme reactions to maintain the sodium potassium differential across a membrane. A proven system is awaiting exploitation.

Chairman Amdahl: Thank you. I now recognize Table 8.

Dr. Moore: The questions that have been raised about "is this really necessary?" and "aren't we peering a little too far into the future?" are, of course, relevant. I mean, if we are talking about a computing installation that does problems of the kind that people do now, there may be reasons why we don't want to go that far in giving the machines this much autonomy or this much self-repairing ability. However, the over-all possibilities of these machines, the inherent capabilities of the kind of things that we can do with logical elements if we get larger and larger aggregations of them, are the things that make it worthwhile for us to consider what would happen if we got so many components together that even the twenty per cent that the marginal checking won't catch would be entirely out of range, and we couldn't handle the troubles by present-day methods.

There do seem to be real problems that present-day methods would *not solve* but, of course, most of today's discussion is not on the present-day level of something that anybody is currently going to build.

With regard to the suggestions that perhaps the quick and easy solution is to duplicate everything and have a monitor element, and when your monitor says anything is wrong throw out the two elements and the monitor—that may reduce errors, but it is still not absolutely foolproof. What happens when the monitor becomes very optimistic and decides nothing is wrong—has a permanent failure resulting in its indicating that everything is o.k.?

Well, this doesn't cause the machine to malfunction yet, but it is then only a matter of time until one of the tubes that you are comparing does fail and then you will run into real trouble.

Chairman Amdahl: I now recognize Table 7.

Dr. Shannon: I would like to comment on several remarks that were made before about the analogy between self-correcting machine and communication theory using redundant elements as a measure to provide greater reliability. This is an analogy which I have thought about quite a bit, and I find it is a valuable one. However, there are two fundamental differences between the two cases. In the first place, in a communication channel, when you introduce redundancy in coding, you assume that at the ends of the channel you have reliable, logical elements to calculate from the perturbed message what the true message was. You assume no errors in these elements.

On the other hand, in a computing machine, if you assume unreliable elements, you must do your logic with these unreliable elements. Thus, the part of the machine that is failing has to be checked with the other parts of the machine that may also fail. It is necessary to introduce redundancy all along the line.

The second difference is that in communication theory one obtains elegant theorems only by going to the limit of using a long delay and arbitrarily involved coding systems. In a computing machine, it is difficult to find any reasonable way to define a limiting process which is equivalent.

I also would like to mention here some work of von Neumann which is quite important in this connection. He assumes a component that is somewhat like a neuron—a kind of idealized, unreliable neuron which will behave tolerably well but give a certain frequency of errors. Von Neumann shows that you make a reliable neuron from a sufficient number of unreliable ones, and that, as a matter of fact, you can approach perfect operation.

The redundancy that is required for approaching high reliability by the methods he uses is very great. In order to start with an element which would have a probability of the order of 1 in 100 of failing, and get up to an element which would be good enough to use in a human brain and last thirty or forty years without a single error, each one would be replaced by about 20,000. In other words, a redundancy of 20,000 to 1.

Chairman Amdahl: Thank you. I will now recognize Table 4 for part of one minute.

Professor Wiesner: In most cases, of course, the information is carried as time modulation, and not as digital numbers. In fact, I don't know if anyone has ever discovered a digital adding unit in the brain, but I think it is very important to realize that these data processing systems are systems in which the neurons behave very much like vacuum tube flip-flop gadgets. The transmission is all or nothing, and the transmission in the nervous system is all or nothing, and when you get gradated action it is because numbers of these impulses add up and work together.

Chairman Amdahl: I am going to have to declare a ten-minute recess, after which those who wish to continue the discussion will get together up in front of this hall, and those who wish to get together in the discussion of germanium diode testing, which will be moderated by Mr. Heath, will get together in the rear of this room. The p.a. system will be off. Thank you!

[After a ten-minute recess, there was further, unrecorded discussion, at the conclusion of which the meeting adjourned at five-thirty o'clock.]

Summary of Diode-Testing Panel Discussion 1954 IRE National Convention*

Participation in the Diode-Testing Panel discussion was limited, in the main, to one diode manufacturer and three computer manufacturers. Since such limited participation does not represent the industry as a whole, the entire discussion has been considered unsuitable for publication. However, a short summary of the discussion is submitted for the record.

The purpose of the panel was to discuss semi-conductor diode-testing as it applied to computers. Some of the testing problems which were discussed were: volt-ampere characteristic testing, sweep versus dc methods, reverse transient response tests, forward transient response tests.

A major problem discussed was specification writing. It was admitted that there has been essentially no attempt at standardization of tests by the computer industry. There is no indication that the ac sweep method will gain universal acceptance as a specification for the volt-ampere characteristic. Reverse transient test methods are in a severe state of confusion. The panel decided that it would like to end the confusion and decided that a round-robin testing procedure would be a good starting place. The forward transient response has not been specified by enough users to yet be a confusing factor. The manufacturer present agreed that test standardization was a desirable thing.

Another major problem discussed was the lack of sufficient data being supplied by manufacturers on components already being produced. It was agreed that if the manufacturers presented any data at all on such things as transient response it would be helpful. Statistical data (of electrical characteristics) on all diode types is singularly lacking, too.

The treatment of these problems was not extensive and the only action taken concerned the round-robin on reverse transient response testing.

* Presented by A. W. Lampe and H. F. Heath, Jr

Contributors

W. J. Cunningham (M'47) was born in Comanche, Tex., on August 21, 1917. He received the A.B. degree from the University of Texas in 1937 and the Ph.D. in applied physics from Harvard University in 1947.

Between 1940 and 1946 he did research and instructed at Harvard in acoustics and electronics. Since 1946 he has been in the department of electrical engineering at Yale University, where he is now an associate professor. His present activity is primarily in the field of nonlinear analysis.

Mr. Cunningham is a member of Sigma Xi, and the Acoustical Society of America.

F. A. Foss (S'43–A'45) was born in Lynn, Mass. on February 19, 1922. He graduated from Tufts College with the B.S. in E.E. in 1944. He received the M.S. in E.E. in 1948 and the E.E. degree in 1950 from M.I.T. While at M.I.T., he was a member of the electrical engineering teaching staff and the Whirlwind computer research staff.

From 1944 through 1946 he served in the Army Signal Corps as a communications and operations officer in the United States and the Western Pacific Area. In 1950, Mr. Foss joined the International Business Machines Corporation, Endicott, N. Y., where he has participated in the design of business machines and military control systems.

Mr. Foss is a member of Tau Beta Pi, Sigma Xi, and the Association for Computing Machinery.

M.W. Fossier (M'53) was born in New Orleans, La. on March 30, 1928. He received the B.S. degree in mechanical engineering at Louisiana State University in 1945, the M.S. degree in aeronautics from the California Institute of Technology in 1946, and the professional degree of Aeronautical Engineer from the California Institute of Technology in 1947.

From 1947 through 1950, Mr. Fossier was employed by the El Segundo Plant of the Douglas Aircraft Company as an aerodynamicist. In 1950, he joined the missile and radar division of the Raytheon Manufacturing Company, where he is presently employed as a staff engineer in the field of guided missiles systems analysis.

Mr. Fossier is a member of the Institute of Aeronautical Sciences.

Howard Hamer (A'50) was born in New York City on May 7, 1923. He received the B.E.E. degree in 1948 from the College of the City of New York, and the M.E.E. degree from North Carolina State College in 1950.

From 1949 to 1951 Mr. Hamer was employed as a servomechanisms engineer by Bell Aircraft Corporation. In 1951 he joined Electronic Associates, Inc., as a project engineer specializing in feedback control systems, in which capacity he has taken part in the development of analog computing equipment.

Mr. Hamer is a member of A.I.E.E., Eta Kappa Nu and Sigma Xi.

G. R. Partridge (S'45–A'51) was born in Denver, Colo., on October 26, 1925. His electrical engineering training was received at Yale University, where he held a Gerard Swope Fellowship, 1946–47, and received the Ph.D. degree in 1950.

During the summers of 1947 and 1948, he was employed in the transmitting and industrial tube section of the General Electric Company. In 1949, he was appointed an assistant instructor in electrical engineering at Yale University, and has remained in the teaching profession except for academic vacations spent at the Radio Corporation of America and the Hewlett-Packard Co. In 1950, Dr. Partridge joined the electrical engineering faculty of Purdue University, where he is presently employed as an associate professor.

He is a member of the AIEE, Tau Beta Pi, Sigma Xi, Eta Kappa Nu, and an associate member of the Acoustical Society of America.

H. A. Rosen (S'48–A'51) was born in New Orleans, La. on March 20, 1926. From 1944 through 1946 he served as electronic technician in the U. S. Navy. He obtained the B.E. degree from Tulane University in 1947 and the M.S. degree from the California Institute of Technology in 1948, both in electrical engineering. In 1951 he received the Ph.D. degree in electrical engineering and aeronautics from the California Institute of Technology.

Since 1948 Dr. Rosen has been engaged in the development of anti-aircraft guided missiles by the Raytheon Manufacturing Company. He is a member of Tau Beta Pi and Sigma Xi.

Bernard Widrow was born in Norwich, Conn., on December 24, 1929. He attended the Massachusetts Institute of Technology where he received the S.B. and S. M. degrees in E.E. in 1951 and 1953, respectively. Since 1951, Mr. Widrow has been engaged in research on magnetic-core memories, first at the M.I.T. Digital Computer Laboratory and later at M.I.T.'s Lincoln Laboratory.

Mr. Widrow is a member of Sigma Xi.

PGEC News

Message from Chairman
Harry T. Larson

The purpose of this message is to describe briefly several PGEC activities which would not otherwise be communicated to the members of the PGEC. It is hoped that this will convey some idea of the current activities of the various mechanisms of the PGEC. The communication of this information to you is coupled with an invitation to participate actively wherever you note something of interest.

Editor for the Transactions—The change of editorship from Dr. Werner Buchholz to Dr. Ralph E. Meagher represents a pioneering step on the part of the PGEC. As chairman of the Publications Committee, Dr. Buchholz carried, without compensation, the enormous load of producing the Transactions and the Computer Issue of the Proceedings of the I.R.E. Through the vigorous efforts of last year's officers, the PGEC obtained approval of the establishment of a paid editorship. This move will, of course, make it possible to expand and improve the services provided by the Transactions. We are particularly fortunate in having obtained the services of Dr. Meagher. It is noted that the Transactions is the most important single activity of the PGEC, in its role as catalyst for the flow of computer information. You are all urged to consider seriously the Transactions as an outlet for papers you publish. The large reviewing staff will continue to filter the papers carefully, so as to maintain a high caliber in the articles which reach the pages of the Transactions.

Size of the PGEC—The paid membership stands at 2,600. This makes the PGEC the largest of the PG's, although it was the sixteenth formed. The size is due in good part to the energetic work of the past PGEC officers and committees and is a symptom of the rapid expansion of the computer field.

Treasury—At the time of this writing, the balance in the treasury is about $13,000.00. Considerably more than desirable, this unusually large balance has built up primarily because nearly every publication and conference undertaken by the PGEC has shown a modest profit. It is not the policy of the PGEC to attempt to make a profit on such activities. The policy is simply to perform a service to the computer industry and try to break even in the process. The profits to date are another symptom of the rapid growth of the computer field, which has led to unexpected attendance at meetings and unexpected demand for publications.

This cash will, of course, be transformed into increased service to the industry and the PGEC membership. Plans for making use of the money include expansion of the Transactions, increased financial aid to PGEC chapters, student activities, and awards. This excellent financial position will make it possible to provide you an increasing return on your investment in the PGEC.

Student Activity—A national Student Relations Committee is being formed, under the chairmanship of Dr. Harry H. Goode, University of Michigan. The activity of this committee to date includes a survey of the universities in the United States, to determine which ones teach digital and analog computer courses, which ones plan to teach such courses, and the level at which such courses are taught. Liaison with these universities has been established via a faculty representative and student representatives. Complementary copies of the Transactions have been sent out. Future plans include supplying student engineering societies with movies, speakers, demonstrations, field trips, awards, etc. Student activity of this type has already developed spontaneously in the New York and Los Angeles areas. In Los Angeles the student relation activity is a cooperative venture including the AIEE and ACM. The IRE Student Quarterly publication prints short articles of interest to engineering students. The editor is particularly interested in "scoop" articles which describe some new device. You are hereby notified of this new channel for publishing the results of your efforts. Please send your articles to Dr. Werner Buchholz, I.B.M.

Constitution Modifications—A considerable amount of work was accomplished last year on several changes in the PGEC constitution. The work was undertaken by D. H. Gridley, Naval Research Labs., and will reach completion this year. Proposed changes stem primarily from suggestions made by the IRE. These suggestions result from IRE's growing experience in administering the Professional Group plan and from the continuing give-and-take between IRE and the PG's. Other changes are being proposed from within the PGEC, resulting from two years of operating experience.

Joint Computer Committee Charter—This committee is sponsored by the AIEE, ACM, and IRE. The PGEC generally acts as the IRE representative on it. The purpose of the organization is to "aid in the promotion of close cooperation and coordination in the activities of the sponsoring societies related to the field of computer engineering and allied arts and sciences." It is this committee which sponsors the two Joint Computer Conferences held each year. In the light of two or three years experience, the charter of the committee has been revised. The revised document is now in the process of being approved or rejected by participating societies. The Administrative Committee of the PGEC is representing IRE in this matter.

Liaison with Industry—Over the past two years a liaison system has been established by the PGEC to set up a channel directly to all organizations engaged in computer activity. This system consists of one "liaison representative" in each organization. The man currently responsible for this system is H. H. Sarkissian, National Cash Register. This system is used for such things as advertising meetings, soliciting papers for meetings or for the Transactions, sampling opinion, and securing new PGEC members. Although the designers of this system have been as thorough as possible in contacting organizations, it is felt that some companies were missed. If you believe that there is no liaison representative in your company, notify Mr. Sarkissian, giving the type of work performed by your organization.

Harry T. Larson
Chairman
IRE Professional Group on
Electronic Computers

Western Computer Conference

The 1955 Western Computer Conference will be held March 1–3 at the Statler Hotel in Los Angeles, Calif. The theme of the conference is "Functions and Techniques in Analog and Digital Computers." Tutorial sessions, discussion groups, and field trips have been planned as well as a series of technical papers. A highlight of the technical program will be a session devoted to operations research. Individuals wishing to present papers to the conference should write to the Program chairman: Dr. G. D. McCann, Engineering Department, California Institute of Technology, 1201 East California, Pasadena, Calif.

Facilities will be provided for exhibits of computer systems or components by manufacturers. Manufacturers desiring exhibit space are invited to write to the Exhibits chairman: Mr. L. L. Kilpatrick, Computer Systems Group, North American Aviation Corporation, 12214 South Lakewood Blvd., Downey, Calif. W. L. Martin and William Gunning are, respectively, chairman and secretary for the conference.

Stanley B. Disson, News Editor
Burroughs Corporation
Research Center
Paoli, Pa.

Joint Eastern Computer Conference

The 1954 IRE-AIEE-ACM Eastern Computer Conference will be held November 8–10 at the Bellevue-Stratford Hotel in Philadelphia, Pa. The theme of the conference will be "Design and Application of Small Digital Computers."

Technical papers, exhibits by manufacturers, and tours of computer installations in the Philadelphia area are planned. The Proceedings of the conference will be published by the sponsoring societies and may be ordered from any one of them. Information on registration, which is open to all, may be obtained by writing: Eastern Joint Computer Conference, P.O. Box 7825, Philadelphia 1, Pa.

Reviews of Current Literature

It is the intention of this section to review articles that have been published since January 1, 1953, and to publish eventually reviews of all books pertaining to the computer field. All articles and books reviewed are numbered sequentially for each year; where known the Universal Decimal Classification number is also given. The editors wish to express their gratitude to the reviewers who, through their efforts, make this section possible.

H. D. Huskey, Editor

GENERAL

54–175

Bibliography on Data Storage and Recording—G. L. Hollander. (*Commun. and Elec.*, No. 11, pp. 49–58; March, 1954.) This bibliography contains 330 items on publications applicable to the fields of data storage, recording, analog-to-digital conversion, data presentation and telemetering. Listing is alphabetical by authors, although a subject index is included. A very brief abstract appears with each item.

R. K. Richards

54–176

Computers and Automation—Reference Information—(*Computers and Automation*, vol. 3; July, 1954.) Two new lists (Nos. 1 and 2 in the following list) have been added to the other reference information maintained by this journal: (1) "Automatic Computers" (cumulative list; 136 entries). (2) "Automation—List of Outstanding Examples" (cumulative; 16 entries). (3) "Roster of Organizations in the Field of Computers and Automation" (supplement in this issue). (4) "Patents" (pertaining to computers and associated equipment; March 23, 1954 to May 11, 1954 in this issue). (5) "Roster of Automatic Computing Services." (6) "Glossary of Terms in the Field of Computers and Automation" (not in this issue). (7) "Roster of Organizations Making Components" (not in this issue). (8) "Who's Who in the Field of Computers" (not in this issue).

G. E. Gourrich

54–177

A Digital-Analog Machine Tool Control System—Harry W. Mergler. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 46–59; April, 1954.) The work described in this paper was undertaken originally to facilitate the fabrication of experimental three-dimensional blades for compressor and turbine rotors. For the two dimensional case, two methods are presented for generating a continuous contour through a set of discrete points defining the desired locus of a cutting tool. The first method employs three mechanical ball-and-disc integrators to pass a third-degree polynomial through four adjacent data points, the center interval being taken as the desired surface. As successive intervals are traversed, the integrator positions are modified by fourth differences read from IBM cards. In the second method, a flexible metal spline is positioned at four points by servos operating from co-ordinate data read from cards. The center interval is then traced by a spark-gap probe. To permit independent operation of either interpolator and the actual machine-tool power servos, a system was developed for recording the interpolator outputs on magnetic tape.

E. C. Johnson

54–178

Resistor Reliability—Whose Responsibility?—J. Marsten. (*Proc. Eastern Computer conf., Joint IRE-AIEE-ACM, December 8–9, 1953 Washington, D. C.*, pp. 109–112; 1954.) The author points out that so-called component failures are often caused by improper applications of components. Several examples of misapplication of resistors are discussed. Generally, in these cases, the cause of component failure is some factor that was never even considered in applying the component to the circuit. The author contends that if the necessity for equipment reliability were formalized in design specifications, design engineers would recognize the fact that component reliability can be achieved by a process of compromise or redesign such as that presently used in achieving low-cost or small-space consumption.

Russell A. Kirsch

54–179

Electron Tube Performance in Some Typical Military Environments—D. W. Sharp. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 77–83; 1954.) Aeronautical Radio, Inc. is conducting an extensive tube surveillance program for the three military services. Of 2,343 tubes returned from a land base only 9 per cent had mechanical defects, 63 per cent fell below electrical limits usually in mutual conductance, while 25 per cent showed no defect. Most of the latter were near the specification limit so they might be the result of disagreement in the calibrations of the tube testers. Some typical rates of deterioration of mutual conductance, in micromhos per 1000 hours, are 6BA6 450, 6SK7 70, 12AT7 700, 6201 350.

From shipborne equipments 22 per cent of the tubes returned had mechanical defects, mostly shorts or opens, 44 per cent were below electrical limits, and 34 per cent showed no defect. The latter were distributed about the same as new tubes so must have been arbitrarily replaced. Failures and removals followed exponential laws rather well with 50 per cent failures expected in 6000 hours and 50 per cent removals in 4000 hours. Rejection is even faster in aircraft environments.

R. D. Elbourn

54–180

Reliability and Its Relation to Suitability and Predictability—E. B. Ferrell. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 9–10, 1953, Washington, D. C.*, pp. 113–115; 1954.) Reliability is the word frequently used in connection with electronic components and systems. This paper points out that the word means different things to different people. In the author's effort to find out what it does mean he finds that it does not have a meaning sufficiently precise to be of much value. The words suitability and predictability come closer to the meaning most people want in discussing electronic components. Using as an example a group of relays all constructed at one time, the author shows that tests just after construction and during the "life" of the relays produce information which may be used to predict the performance of other groups under suitable circumstances. It is this predictability of the characteristics of a component throughout its life which is so important to the circuit engineer. The paper closes with a plea to producers of components to make products predictable.

R. E. Meagher

ANALOG SYSTEMS RESEARCH

54-181

Harmonic Cancellation From Computing Voltage Source in Servo Analogue Computers—James Alman. (*Elec. Eng.*, vol. 73, pp. 711–713; August, 1954.) Servo systems using an ac carrier frequency become subject to disorders caused by harmonic voltages appearing on the computing voltage. This article describes a system for canceling out these harmonic voltages in the summing network of the servo amplifiers. This is accomplished by the injection of harmonics into the quadrature rejection circuit of the servo system. The article consists primarily of a mathematical development supporting the system described. Simplified block diagrams and circuit diagrams are included.

Harry T. Larson

ANALOG EQUIPMENT

54-182

Analog Computer for the Roots of Algebraic Equations—L. Löfgren. (Proc. I.R.E., vol. 41, pp. 907–913; July, 1953.) By manipulating the algebraic equation, whose roots $z_1, z_2, \cdots z_n$ are sought, a pair of equations, expressing the condition upon the real and imaginary components of the original, result. In these derived equations, exponents of the variable z appear as multipliers of a basic frequency component in sinusoidal terms. Starting then with an oscillatory and frequency dividing chain, the derived equa-

This page has been left blank in order that readers may mount all reviews on cards.
—*The Editor*

tions are assembled term by term, with provisions for entering the coefficients of the original equation made at this point. The simultaneous solution of the derived equations is detected by a gating circuit which indicates the corresponding value of the root on the cathode-ray presentation; the latter then presents the complex plane with roots appearing as spots. The author discusses a normalizing procedure for improving accuracy. Although the number of electronic tubes exceeds 200 (for equations up to eighth degree), its computing speed is high enough to allow for continuous coefficient variations and their resulting root-loci to be observed.

L. G. Walters

54-183

A Transmission-Loss Penalty Factor Computer—C. A. Imburgia, L. K. Kirchmayer and G. W. Stagg. (*Elec. Eng.*, vol. 73, pp. 649–652; July, 1954.) As electrical power distribution systems expand, the proper coordination of incremental fuel costs and incremental transmission losses becomes increasingly important. A major step in the solution of this problem was the development of the "incremental cost slide rule," which provided a means of scheduling generation on an incremental fuel cost basis. The computer described in this article is used in conjunction with this slide rule. The computer is a special purpose machine which will calculate transmission-loss penalty factors used to adjust the relative positions of "incremental fuel strips" on the slide rule. The system of differential equations solved by the computer is described. The description of the computer itself is extremely brief, and no information is given on its internal structure. The procedure for the combined use of the computer and the slide rule is given.

Harry T. Larson

54-184

Network Analyser of the Electrical Research Association—(*Nature*, vol. 174, pp. 214–215; July 31, 1954.) "The network analyser constructed by the Electrical Research Association, the first completed section of which was exhibited to the technical and scientific press on April 28, has been designed for application to the widest possible range of problems. To this end the equipment can operate at any frequency from 160 c./s. to 16 kc./s., so that frequency can be a variable or one of the unknowns in a problem. The maximum accuracy is at the "base" frequency of 1,592 c./s. This versatility is accompanied by the possibility of extension to a size not less than that of any existing analyser. The equipment is believed to be unique in providing steady-state, transient and harmonic solutions on the same installation." There is a selection system to the terminals of the central measuring equipment, indicating simultaneously current, voltage, load, and reactive load at the selected point. "Physically small components used in the network elements yield an inexpensive and compact design with short interconnecting leads and thus increased frequency-range. A resolution and repeatability of 0.1% of amplitude (and 0.1° of phase angle at amplitude levels greater than 10%) has been aimed at, with an absolute accuracy of 1% and 1° at the base frequency. Although the equipment is for use primarily in the service of the electrical industry, it is intended that so far as possible its versatility and flexibility shall be made available for the solution of problems within its range irrespective of their origin."

Donald E. Hart

UTILIZATION OF ANALOG EQUIPMENT

54-185

Scale Factors for Analog Computers—James B. Reswick. (*Prod. Eng.*, vol. 25, pp. 197–201; March, 1954.) A technique is described which permits a simple visualization of the scale factor problem in analog computers. The method allows time scaling to be considered apart from scale factor determination. First a true time analog of the system being studied is set up. Scale factors between the true-time analog and the physical system are determined. A nondimensional analog is to be avoided since difficulties arise when nonlinearities are present and because the physical significance of parameters is often lost. A block diagram of the true-time analog can be set up in a plane parallel to the diagram of the physical system and the proportionality constants between the electrical parameters and the physical parameters determined. The determination involves setting the coefficients in the true-time analog so that they are within the range of the computer. This can usually be done by the consideration that all coefficient settings should be close to unity. If the dynamic range of the computer is greater than or equal to the dynamic range of the physical system, this can be done. If it cannot be done, there is a strong possibility that the problem is formulated in a manner inconsistent with "engineering accuracy." Second, the true-time analog is transformed to a time scale convenient for the computer by a linear transformation of all frequencies and time constants. A third block diagram may then be drawn, parallel to the first two, showing the correspondence between the fast or slow-time analog, the true-time analog and the physical system.

T. H. Bonn

DIGITAL COMPONENT RESEARCH

54-186

Electron Tube and Crystal Diode Experience in Computing Equipment—J. A. Goetz and H. J. Geisler. (*Proc. Eastern Computer Conf., Joint IER-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 67–72; 1954.) Records have been kept on forty-five tube types and three crystal diode types used in many IBM computers. Tubes failing in service are given elaborate tests in a central laboratory, and the data punched on cards. Application memoranda are based on this experience. Tube types 5844, 1684, and 5965 have been developed as one result. Statistics are given for failure rate and per cent survival. At 5000 hours the tube survival ranges from 93 per cent to 97 per cent for types shown. Diodes lasted as well as the best tube type. The authors do not distinguish between replacements due to routine inspection, general trouble-shooting, and component responsibility for computer shutdown. No discussion is given of apparatus design concepts and detail engineering which can strongly affect component reliability, but types of tube deterioration are listed which cause trouble in sensitive circuits.

J. Howard Wright

54-187

Magnetic Ferrites: New Materials for Modern Applications—V. E. Legg, C. D. Owens. (*Elec. Eng.*, vol. 73, pp. 726–729; August, 1954.) This article surveys, from the designer's point of view, properties of the available ferrites which have reached commercial use. Also brief descriptions of a variety of applications of ferrites.

Harry T. Larson

54-188

Circuits to Perform Logical and Control Functions with Magnetic Cores—S. Guterman, R. D. Kodis and S. Ruhman. (*Convention Record of the I.R.E., 1954 Nat. Conv.*, Part 4, pp. 124–132; 1954.) Magnetic core logical circuits based on the "single-line" shift register are described. A "single-line" shift register can be constructed from cores each having an input, output, and shift winding. The shift windings of all cores are connected in series, and the output winding of each core is connected to the input winding of the next through a delay network. A current pulse applied to the shift line resets all cores to the "zero" state and induces a voltage in the output windings of all cores which were previously in the "one" state. A "one" read out in this manner, after passing through the inter-core delay network, is written into the succeeding core, so that each shift pulse advances the information in the register one stage. By feeding the output of one core to the inputs of two or more cores a BRANCH is formed. To obtain an OR function, the input of one core is connected to the outputs of two or more cores. The INHIBIT function is produced by canceling a "one" in the input winding of a core by a "one" applied simultaneously to an oppositely wound fourth winding. The shift register, BRANCH, OR, and INHIBIT circuits are the building blocks from which more complex logical circuits are assembled. Several such circuits are described, including pulse pattern generators, counters, adders, subtractors, gating and extracting circuits, a multiplier, and a divider, all working with binary information in serial form.

Gerhard Walter

54-189

Recent Progress in the Production of Error-Free Magnetic Computer Tape—J. C. Chapman and W. W. Wetzel. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 102–104; 1954.) Defects in magnetic tape cause both stray noise pulses and signal dropouts, the latter being much more frequent. Noise pulses can be caused by magnetically hard inclusions, or pinholes, or streaks in the coating. Pinholes and streaks can also cause signal dropouts but the usual cause is a "nodule" protruding from the tape surface so that it lifts the main body of the tape away from the gaps in the recording and reproducing heads. Investigation showed that the nodules were caused by oxide clumps, acetate particles, filter fibers,

This page has been left blank in order that readers may mount all reviews on cards.
—The Editor

miscellaneous particles, tape creases, and defects in the backing caused by dents in the wheel on which the plastic film was cast. Efforts to improve computer-grade tape reduced the errors from 3.25 per roll of ½ inch by 2,400 feet in March 1953 to only 0.18 per roll in August 1953. Thus one may now expect over 75 per cent of untested rolls of computing tape to be error-free.

R. D. Elbourn

54-190

Transistors Amplifiers Reduce Delay Line Attenuation—A. H. Schooley (*Electronics*, vol. 27, pp. 181–183; May, 1954.) This article described the results of an experimental study of a nine section, 70 microsecond delay or storage device composed of commercially available distributed constant delay line. The overall attenuation of the 70 microsecond delay line is reduced to 0 db by placing transistor amplifiers between sections. The characteristics of the individual sections were as follows: characteristic impedance, 1,000 ohms; delay time, 7.8 microseconds; band pass, 0 to 2 mc. The pulse width used in the tests was 10 microseconds with build up time and decay time about 0.05 microsecond each. After passing through nine sections of delay line (70 microseconds) and nine transistors amplifiers the build up and decay times were about 0.9 microsecond each. The author states that these results were obtained even though the input and output impedances of the transistor amplifiers did not match the delay line characteristic impedance and thus, caused small undesired signals due to the mismatch. Transistors used were type 1698. Results of measurements showed that less amplifier gain would be required if a better match were obtained between delay line and amplifier.

Norman F. Loretz

54-191

Transistor Shift Registers—C. Huang, E. Slobodzinski and B. White. (*Convention Record of the I.R.E., 1954 Nat. Conv.*, Part 4, pp. 140–144; 1954.) This paper describes a type of shift register which uses two point contact transistors per stage. Circuits used for the memory and temporary storage functions are described and three shift registers which have been built using these circuits are shown. Two of the registers, which have one source of advance pulses, operate up to about 200 kc. The third register has two sets of properly phased advance pulses and faster transistor circuits. It can operate at 1 mc. Block diagram-wise, these registers are conventional. Circuit-wise, they have some interesting features, such as the use of an inductance and diode in the base circuit of a bistable flip-flop to improve the switching sensitivity. Interstage coupling using capacitors is analyzed qualitatively, and transistor requirements are stated. Good reliability is claimed but no voltage margin or life data are included.

E. L. Younker

54-192

Packaged Logical Circuitry for a 4-Mc Computer—Norman Zimbel. (*Convention Record of the I.R.E., 1954 Nat. Conv.*, Part 4, pp. 133–139; 1954.) Methods for handling the problems introduced by the use of crystal diode gating structures at high repetition frequencies are discussed. It is shown that through the use of suitable circuit techniques a reduction of forty per cent in the gate currents, compared to conventional diode gates with the same logical configuration, can be achieved. Gate signal transfer efficiency of ninety per cent may be realized, and at the same time a tenfold decrease in gate junction resistor power is possible. The design of one unit for a packaged circuitry computer is developed in some detail. Design methods are presented which overcome these typical problems: the effects of the appreciable current—reversing switching time of gate and clamp diodes; the isolation of the shunt capacity of the load gate junction from the gate driver so that a high gate transfer efficiency can be attained. A series peaking inductance is used which, with the shunt capacity of the load gate, forms a single section delay network. Compensatory delays are provided. The reshaping of the pulses passed through the gate network is accomplished by coincidence with and logical multiplication by a standard reshaping, or clock pulse, which feeds a regenerative feedback amplifier that has a low impedence transformer output capable of driving ten standard gate legs. Circuit diagrams are included and timing charts are employed to compare performance characteristics.

Richard H. Baker

54-193

A Commutator Switch Derived from Binary Scalers in Cascade—R. Parshad and A. Sagar. (*Rev. Sci. Instr.*, vol. 25, pp. 395–396; April, 1954.) Circuits are described which make it possible to get a voltage at a different point in space for each configuration of a cascaded binary scaler or a decade scaler. The method involves connecting the plates of the nonconducting halves of the flip-flops in a given configuration to the cathodes of a diode system. The plates of the diodes are tied together and returned to a 250-volt supply through a resistor. The resultant output voltage coincides in duration with the flip-flop configuration. Though it can be worked out from the circuit diagram, an explicit statement that the left half of a flip-flop conducting means zero, that the least significant digit appears on the left, and that the binary count begins with 1 and ends with 000, would have clarified the table on which the presentation is based.

A. J. Dowling

681.142:621.3(084.2) 54-194

Electronic-Circuit Technique for a High-Speed Computer—G. Piel (*Onde élect.*, vol. 34, pp. 38–46; January, 1954.) (Courtesy of Proc. I.R.E. and *Wireless Eng.*) Commonly used computer-circuit units based on a triode or Ge diode are represented by a symbol indicating the unit and its mode of operation. The functions of different combinations of the basic units are listed and their application in a serial binary-scale computer is illustrated using the notation described.

54-195

Electrostatic Reading of Perforated Media—Samuel Lubkin. (*Convention Record of the I.R.E., 1954 Nat. Conv.*, Part 4, pp. 106–108; 1954.) Various methods which have been used to read holes punched in tapes and cards are described. The author then presents a new scheme for electrostatic reading of holes in tape or sheet storage media. The scheme uses capacitive coupling of a high frequency signal between two electrodes separated by the storage medium. The tape or sheet must act as an electrostatic shield and thus allow coupling between electrodes only when holes are present. The high frequency signal coupled to the pickup electrode must be amplified and detected to show the presence of a hole in the storage medium. Problems of shielding and coupling are noted. Various configurations of mechanical construction to realize serial, parallel and series-parallel reading of holes are proposed. Results of tests, which are presented, indicate that the proposed scheme is feasible, so far as amplitude of coupled signal and cross talk between pairs of electrodes are concerned. The author suggests that sharp rise pulses can be used, if timing permits, in place of the high frequency signal, however, the attenuation will be more severe. This method of reading perforated media is to be used in conjunction with a computer being developed by the Underwood Corp.

Norman F. Loretz

DIGITAL SYSTEMS RESEARCH

54-196

Elimination of Waiting Time in Automatic Computers with Delay-Type Stores—A. L. Freedman. (*Proc. Camb. Phil. Soc.*, vol. 50, part 3, pp. 426–438; July, 1954.) This paper presents a very thorough study of the effectiveness of several different engineering and programming devices for reducing the waiting time involved in the use of delay-type stores (including magnetic drums). Many different methods, often described by such general terms as "optimum coding," are here carefully distinguished and studied in relation to various machine operations. The discussion is limited to single-operand-address machines (without address of next instruction) and the results are based on a survey of EDSAC programs. The value of the paper lies mainly in serving as a model for investigations of this kind, and in bringing out the factors which influence the over-all speed of a computer. The author concludes that of the methods considered, those providing greater economy of waiting time also require greater complexity of machine or program, with the possible exception of the provision of temporary storage registers. He is of the opinion that these methods are not justified unless the waiting time is at least half the total time.

S. Gill

54-197

Digital Computers for Linear Real-Time Control Systems—Ralph B. Conn. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 33–37; 1954.) The mechanization of a special-purpose computer is discussed which could be useful in certain control applications. The computer would be specialized to combine samples of variables in a linear manner; i.e. to solve the basic equation

$$o_k = \sum_{j=0}^{M} a_j u_{k-j}$$

This page has been left blank in order that readers may mount all reviews on cards.
—The Editor

where u_{k-j} are samples of a variable, 0_k is a computed sample, and a_j are constant coefficients. More than one input variable and/or previously calculated samples of the output variable could also be included in the linear combination, which could make the computer generally useful in control applications. In the simplest form of mechanization the a_j's are assumed to be negative powers of 2 so that each term of the equation above would involve a shifting operation and an addition. The arrangement of the drum storage would make the indicated sequencing automatic. This saves a lot of equipment at the expense of specialized purpose. Extension of the mechanization to the more general case of unrestricted a_j's is discussed. It is not clear whether the computer was built or not, but apparently it was designed in sufficient detail to permit comparison of equipment with other types of computers. For the special purpose stated the special computer is appreciably more efficient.

John M. Salzer

DIGITAL EQUIPMENT

54-198

Automatic Inventory System for Air Travel Reservations—M. L. Haselton and E. L. Schmidt. (*Elec. Eng.*, vol. 73, pp. 641–646; July, 1954.) This is a good general engineering description of the American Airlines passenger inventory machine. The machine maintains inventories of available seats on 1,000 flight legs per day for a total of 12 advance days. It is tied to 200 local and remote input-output keysets. Agents can (a) interrogate the inventory machine to determine seat availability, (b) sell seats, causing reduction in inventory, and (c) cancel reservations, causing an increase in inventory. A listing is given of design features incorporated for the prevention of interruptions to service and for the detection of errors or potential equipment failures. Technical design data is included, and a description of "novel operating features" is presented. During 1953, the equipment operated 99.8 per cent of the scheduled time. This reliability points to wider use of such equipment in commercial and military applications.

Harry T. Larson

54-199

An Experimental Digital Flight Control System—Maier Margolis and Eric Weiss. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 23–37; April, 1954.) The mechanization of a breadboard computer is described which served the purpose of proving out the feasibility of digital control of flight. The article starts with an analysis of the flight control problem, which leads to a set of equations to be solved in the computer. The problem seems to involve about 15 to 20 multiplications, the same number of additions, and about a dozen decisions. The resulting computer resembles a general-purpose computer having a fixed program. It uses a magnetic drum for storage of constants and for various delays of several words. Supposedly the drum is only needed for flexibility during the experiments and could ultimately be discarded. Thirteen shaft inputs are used with reflected binary code conversion. Then, starting with the least significant digit first, a reflected to straight binary code conversion is accomplished for each input in two word times: in the first word time the number of ones in the number are counted, while in the second word time the conversion is performed. Counting the ones during the first word time eliminates the necessity of starting with the most significant digit. The exact mechanization of the computer is not given; such questions as how multiplication is performed are not answered explicitly. The computer is the serial type and the fixed program seems to be so arranged that many of the operations overlap in time and the arithmetic and logical equipment are efficiently used. The computer produces a single output with limits. This output controls the pitch channel. The conversion of the seven-digit error is performed by a voltage ladder network.

John M. Salzer

54-200

Design Features of the JAINCOMP-C and JAINCOMP-D Electronic Digital Computers—Donald H. Jacobs. (*Convention Record of the I.R.E., 1954 Nat. Conv.*, Part 4, pp. 98–104; 1954.) The major part of this paper contains a description of the JAINCOMP-C computer designed for a specific real-time application involving nine variable inputs and three controlled output devices. The program performed is repetitive with a 0.1 sec. cycle; the program is controlled by means of an electronically scanned, changeable punched card. Ordinarily three-address instructions are used. A program card contains 128 twenty-four bit instructions. Three additional punched cards hold seventy changeable computation constants. Intermediate results are stored in twenty-four magnetic core registers with eight microsecond access time. The register length throughout is twenty-four bits. Each core is operated as a saturable reactor with only one winding with many turns. Reading is performed by observing the voltage developed across this winding when a reset pulse is applied. Time discrimination is used to distinguish between a one and a zero. Although JAINCOMP-C may perform a predetermined number of iterations, for example, by means of jump-orders it appears to have considerably less flexibility than the JAINCOMP-D. The latter incorporates a 1024-word magnetic memory of the same type as that employed in the C model. JAINCOMP-D is also capable of modifying its own instructions. Both computers have multiplication and division times below 400 microseconds. They are packaged and of table-top size. Automatic self-checking routines are incorporated.

Torben Meisling

54-201

Automatic Measurement of Star Positions—J. Lentz and R. Bennett. (*Electronics*, vol. 27, pp. 158–163; June, 1954.) This article describes a machine developed by IBM to automatically measure and record the location of star images on a photographic plate. The approximate location from previous measurements is entered into the machine from an IBM card. The machine then positions a photoelectric sensing device to the approximate location and proceeds to determine the new exact position. This new location in rectangular co-ordinates is punched onto an IBM card for storage until needed. The locations of star images are measured to 0.2 micron. This automatic machine will measure in one day with greater accuracy as many star image locations as an operator can do visually in a week.

Norman F. Loretz

UTILIZATION OF DIGITAL EQUIPMENT

54-202

Computers: 1954—Mina Rees. (*Sci. Mon.*, vol. 79, pp. 118–124; August, 1954.) An account of the present state of the art "in the technology of construction and in the science of use of digital computers." General descriptions of the present-day large machines are given, and some of the most difficult problems attempted to date are described. Notable among these are the problems of weather forecasting, language translation, and large scale data handling. In the field of data handling, consideration is given to the problem of reorganization of filing methods to take full advantage of a computer's capabilities. The question of the legal validity of documents recorded on magnetic tape and other forms of storage is also discussed. In surveying the future, the author points out the possibilities for higher speeds with smaller computers utilizing magnetic core storage, transistors, and the many miniature components becoming available.

Vernon C. Kamm

54-203

Trends in Electronic Business Data Systems Development—D. E. Wooldridge. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 16–22; April, 1954.) The paper is taken from a luncheon speech and, hence, it is not to be expected that it will say very much or be very profound. It does, however, provide several cautions to the business executive or the electronic engineer who would venture into the business data handling field. Some of these are worth restating, as follows: (1) The systems analysis should be competently and thoroughly done before important decisions are made either on the problems which are to be solved, or the equipment and components to be used. (2) Business data processing is more complicated than often suspected. The reviewer can confirm this, for example, by the approximately 400,000 computer instructions which have been programmed to compute the factory pay roll on the UNIVAC at the General Electric Co. One should be conservative and make liberal use of the "von Neumann constant" in planning business applications. (3) Electronic engineers may tend to give only superficial or incorrect consideration to the business problems, often because of lack of familiarity with these problems. (4) The systems analysis team must be certain that the right problems are being handled. The speaker gives the example of report prepa-

This page has been left blank in order that readers may mount all reviews on cards.
—The Editor

ration; if a high-speed random access machine is used, few reports may be actually necessary. The reviewer adds his experience that, however, it may be possible to let the data system sort out "exceptions" automatically, and present these as abbreviated reports. This leads to efficient "management by exception." (5) Many legal problems have not been decided as yet, or are not uniform, such as the need for recording a signature on retail purchases. The reviewer adds others: Will magnetic tape records satisfy legal requirements for auditing purposes, etc.? In general, the tone of the speaker is to caution persons who would proceed with application of business data systems. The reviewer cannot agree fully with the degree of conservatism the speaker advocates. There are many companies now producing fine equipment that will perform very satisfactorily and that will pay its way in direct, realizable savings. The actual experience of using the equipment then leads to insights not available even to the most thorough systems analysis. There are limits as to how far a systems analysis should go before one proceeds actually to try his hand. Hindsight is always better than foresight, but one does not get hindsight by waiting until the ultimate in equipment and systems has been developed at some indeterminate future date. For example, many billions of miles will be traveled before the ultimate roadable-airplane is available for all-purpose family transportation. So, also, will many excellent applications be made before the all-purpose integrated business-data system is designed.

M. E. Salveson

54-204

Computers in Business—Lawrence P. Lessing. (*Sci. Amer.*, vol. 190, pp. 21–25; January, 1954.) A basic nontechnical description of digital computers. Illustrations of business possibilities for computers are given. An inventory control machine called "Distribution" built by Engineering Research Associates did an inventory job with ten operators that would have required 150 clerks. The revolutionary nature of the big computer is the magnitude of its stored program operation. The memory of these big computers enables them to combine many steps or to take on a series of different problems in any sequence. Prices of $800,000 to $1,000,000 are quoted, and about $10,000 a month is given as the cost on a rental basis.

J. A. Fingerett

54-205

The Automatic Handling of Business Data—Oliver Whitby. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 75–79; April, 1954.) The author points out the possibilities of applying electronic computers to processing business data and the problems confronting the designers and users of such equipment. Business data handling has been increasing in volume to such an extent that costs for handling it manually are prohibitive. Still, the companies cannot afford not to do this work. Electronic computers should afford a cheaper means for doing the work now being accomplished by hand, with fewer errors. In addition, certain other jobs such as sales analysis can be done where it is not now being done. Some applications where computers are already being used are payroll calculations, actuarial calculations, and inventory control. Some problems confronting business are changes in systems to facilitate the use of automatic computers rather than hand methods, and reduction in printing requirements, much of which is "illusory." Design problems include development of random access storage systems and inexpensive methods of recording information into computer language at the source. A brief inventory control example is given in conclusion.

Lowell S. Michels

54-206

A Merchandise Control System—William L. Martin. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 184–191; April, 1954.) This paper deals with possible computer applications in the field of inventory control and data processing. Both the advantages and problems of such operations are mentioned. As an illustration of a possible application, an electronic system is set up for a theoretical mail-order house. The flow of information through such a business is outlined along with the basic electronic units required to do the job. Essentially, the system consists of an Inventory Computer which would control sales, provide current inventory and sales information, and feed pertinent data into a second computer called the Management Computer. This unit would process the information made available to it by the Inventory Computer and provide management reports. Flow diagrams of both the information flow and electronic system accompany the explanation. The paper closes with a brief discussion of the actual equipment that would be needed to establish such an operation.

Gerald Licht

54-207

Production Control With the Elecom 125—Norman Grieser. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 163–171; April, 1954.) This article discusses the background, purpose, aims, cost, actual hardware, requirements, advantages, and disadvantages to the Elecom 125 Electronic Business System. Summing up the hardware in the Elecom 125 system, the author concludes: "Input Tapewriters, producing hard copy and reels of paper tape on which the coded information is contained in the form of printed dots. The central computer, a 1,000 word magnetic drum electronic computer of moderate speed, with associated magnetic tape drives, and facilities to handle paper tape input photoelectrically, with a printer at the output end which produces reels of paper tape containing printed dots. The electronic sorting unit, independent of the electronic computer, with four associated magnetic tape drives. The output units—Underwood electric typewriters with photoelectric readers attached. The photoelectric readers read the printed dots and activate the typewriter to produce hard copy at the rate of ten digits per second. The cost of the systems, depending upon the number of input and output units, is of the order of $175,000 to $200,000."

Donald E. Hart

54-208

Experiments with a Digital Computer in a Simple Control System—T. J. Burns, J. D. Cloud and J. M. Salzer. (*Trends in Computers: Automatic Control and Data Processing, Proc. Western Computer Conf., Joint AIEE-IRE-ACM, February 11–12, 1954, Los Angeles, Calif.*, pp. 60–74; April, 1954.) In the forward loop of an experimental control system containing a simple lag, the authors have inserted a sampler and a digital computer with a time delay. They have demonstrated that the delay and sampling cause the performance of the system to deteriorate in the manner predicted by theory. Within limits, the performance can be restored by insertion of the proper anticipatory computation program in the digital computer. Coefficients of the difference equation for anticipation were computed theoretically and adjusted experimentally so as to produce optimum step function response. These experiments may prove to be the beginning of an interesting new field of application for digital computers.

C. A. Piper

54-209

Computer Applications in Air Traffic Control—Vernon I. Weihe. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 18–21; 1954.) In this paper the author makes an urgent plea for constructive action in the application of automatic data handling techniques to air traffic control, describes typical activities in air traffic control operations as they are now conducted, indicates shortcomings of various proposals that have been advanced, and suggests criteria for realistic approaches to the solution of this important problem. The need is urgent because rates of closing and speed differentials between aircraft have already outdistanced human capability using today's manual methods. An acceptable plan for automatization should provide proper balance between requirements for communication, requirements for computation, human engineering and economic factors. Results should be accomplished in a manner compatible with current operations yet capable of evolution towards fully automatic control as ground and air elements of the system are progressively installed. The SC-31 "guide plan" provides a starting point for a vigorous wakening of interest in a problem so vital to the national interest.

Mary E. Stevens

54-210

A Review of ORDVAC Operating Experience—Charles R. Williams. (*Proc. Eastern Computer Conf., Joint IRE-AIEE-ACM, December 8–10, 1953, Washington, D. C.*, pp. 91–95; 1954.) This report on the performance of ORDVAC, one of three large scale computers at the Aberdeen Proving Ground's Ballistic Research Laboratories, covers a period from March, 1952 to October

This page has been left blank in order that readers may mount all reviews on cards.
—The Editor

1953. One of two graphs gives the week by week performance of the machine, and the other shows the average distribution of time per week in the categories of scheduled and unscheduled engineering, code checking, problem production, idle, available, and standby time. The daily schedule is described in some detail. Three periods are set aside each day for checking the machine. A read around test for the electrostatic memory and a "leap frog" test for the arithmetic section are the principal means of checking out the ORDVAC. Some modifications of the original tape reader and printer have been made, and card handling equipment has been added. As is usual in such cases, the faster card reading system has supplanted the tape to a large extent. The electrostatic memory used a 2 dot system with a read around ratio of 10 to 1, as originally built. Improvements were made in June, 1953, by a change over to a 3 dot system, and, with more reliable cathode-ray tubes, the read around ratio is now 100 to 1. The report concludes with a discussion of preventative maintenance. Block replacement of tubes has been the practice since September, 1952, and the gradual increase of available time on the ORDVAC is believed to be due principally to this. Some trouble was experienced with groups of bad solder joints which have been discovered from time to time. In addition, dust filtering through the cooling system caused one serious short circuit in the high voltage system in the summer of 1952, the recurrence of which was prevented by rearrangement of wiring where possible. The author feels that a continued improvement in the performance record may be expected as the operators become more experienced.

F. H. Hollander

ORIENTATION READING

54-211

What Is a Computer—Neil Macdonald. (*Computers and Automation*, vol. 3, pp. 14–17; July, 1954.) In order to aid the beginner who wishes a simple and correct explanation of basic facts about computers, the author suggests that *Computers and Automation* from time to time print such basic information, bringing it up-to-date with each printing. As a start the author submits general answers to the following questions on an elementary level: (1) What is a computer? (2) What are reasonable operations? (3) What is information? (4) How does a machine take in, record, and remember information? (5) How does a machine manipulate information? How does it actually compute? (6) What is an automatic computer?

G. E. Gourrich

54-212

Human Factors in the Design of Electronic Computers—John Bridgewater. (*Computers and Automation*, vol. 3, pp. 6–7, 10, 17; July, 1954.) The operation of electronic computers still requires considerable contact with human beings. There are two ways to improve human performance in this man-machine relationship: (1) Select and train operators more carefully, and (2) engineer the machine more carefully. The latter is called "Human Engineering." In designing a data processing machine, it is first necessary to determine which tasks should be assigned to the machine and which tasks should be assigned to the human being. Second, the machine must be designed for efficient contact with the human operator, considering such items as: The design of input documents, input equipment, machine indicators and controls; the layout of the operator's workplace and accessibility of the control and indicator panels; ease of required materials handling by the operator, and ease of maintenance by the engineer; control of heat and noise; computer appearance. A further introduction to human engineering is provided by a bibliography of seventeen references.

G. E. Gourrich

BOOK REVIEWS

54-213

Punched Cards—Robert S. Casey and James W. Perry (Reinhold, New York, N. Y. Eds. viii+506 pp., Illus.; 1951.) This book is a compilation, having no less than 36 authors, including the editors. Its function is to describe the use of punched cards in receiving coded information for mechanical sorting, tabulating, totaling, and calculating. There are four main divisions. The first includes an introduction on fundamentals, a treatment of hand-sorted cards and their operation, and a chapter on machine-sorted cards describing cards and machines commercially available. A brief account of some developments in electronic computation is included. The second division is one of "case-history" accounts. The third includes practical suggestions on coding and its mathematical possibilities. Several possibilities in application are shown, similar to some in the second division. A chapter on correlation analysis develops use of punched cards in accumulating sums of products, reducing labor when there are a large number of cases. A final chapter discusses analytical applications and mathematical relations. The fourth division sets out potential future uses in studying scientific literature, and is followed by a bibliography of about 300 titles (other references also appear in the text) and subject and author indexes. The applications shown include condensation and rapid use of files, chemical and physical classification of compounds, patent searching, bibliographical work, use of medical case records, and recording of industrial operations. The widespread present use of such methods in accountancy is not stressed. Numerous pictures and illustrative examples are provided, and coding is described for a number of cases. An interesting feature is the sketch of the work of each author preceding his contribution. The authors include workers in industry, science, medicine, education, libraries, etc., with the first group predominating. The list does not include ranking figures in the field of statistical analysis. *Punched Cards* does not go far into recent advances in extremely rapid calculation, nor into present uses of punched cards in analyzing experimental and survey data. Outlining could be improved. The book will be useful, however, to those who wish to become acquainted with punched-card techniques, and the time-saving in analysis they make possible.

F. M. Wadley

This page has been left blank in order that readers may mount all reviews on cards.
—*The Editor*

Transactions of the I.R.E. Professional Group on Electronic Computers

Index to Volume EC-3—1954

Contents

Index to Authors

Numbers refer to index numbers in contents listing

Index to Technical Subjects

Nontechnical Index

INSTITUTIONAL LISTINGS

MAGNETICS RESEARCH COMPANY
142 King Street, Chappaqua, New York

Magnetic Shift Registers, Magnetic Switching Systems, Pulse Transformers

RAYTHEON MANUFACTURING COMPANY
Waltham 54, Massachusetts

Data Handling & Computing Systems, Magnetic Components, Tape Drives, Computing Services

THE W. L. MAXSON CORPORATION
460 West 34th St., New York 1, N. Y.

Research, Development and Production Computers, Radars, Servos, & Systems

REEVES INSTRUMENT CORPORATION
215 East Ninety-first St., New York 28, N. Y.

Electronic Analog Computers - Servomechanism Components - Resolvers - Gyros - Gears

MID-CENTURY INSTRUMATIC CORP.
611 Broadway, New York 12, N. Y.

Analog Computers - Electronic Function Generators - Recorders - Servomechanisms

REMINGTON RAND, INC.
315 Fourth Avenue, New York 10, N. Y.

Univac and E. R. A. Systems . . . Computing Centers . . . and Training Courses

POLYTECHNIC RESEARCH & DEVELOPMENT CO., INC.
55 Johnson Street, Brooklyn 1, New York

Magnetic Amplifiers for Servo Systems, Voltage Regulators, Speed Regulators

TECHNITROL ENGINEERING COMPANY
2751 No. Fourth St., Philadelphia 33, Pa.

Digital Computers, Memories, Pulse Transformers, Delay Lines

THE TELEREGISTER CORPORATION
443 Fairfield Avenue, Stamford, Connecticut

Development—Data Handling and Inventory Systems—
Digital and Analog Computers

(Please see outside back cover for additional listings.)

INSTITUTIONAL LISTINGS

The IRE Professional Group on Electronic Computers appreciates the support given by the organizations listed below. The listing of products and services is limited in length and therefore is not necessarily all-inclusive.

BELL TELEPHONE LABORATORIES, INC.
463 West St., New York 14, N. Y.

Research and Development for the
Bell System and the Armed Forces

BURROUGHS CORP., Electronic Instruments Division
1209 Vine Street, Philadelphia 7, Pa.

Pulse Control Equipment and Systems; Special-Purpose Tubes;
and Computation Services

COMPUTING DEVICES OF CANADA LIMITED
P. O. Box 508, Ottawa, Ontario, Canada

Digital & Analog Computers, Automatic Control Devices,
Servomechanisms, Research

ELECTRONIC ASSOCIATES, INC.
Long Branch, New Jersey

Analog Computers, Computer Components, Vari-plotter
Plotting Boards, DC Resolvers

FAIRCHILD CAMERA AND INSTRUMENT CORP.
POTENTIOMETER DIVISION
225 Park Ave., Hicksville, L. I., N. Y.

Linear and Non-Linear Precision Potentiometers
Single, Multi-Turn, Wirewound and FilmPots

HUGHES RESEARCH AND DEVELOPMENT LABORATORIES
Culver City, California

Electronic Equipment — Radar,
Guided Missiles, Computers

INTERNATIONAL BUSINESS MACHINES CORP.
590 Madison Ave., New York 22, N. Y.

Electronic Computers,
Scientific Computing Service

LIBRASCOPE INCORPORATED
1607 Flower Street, Glendale 1, California

Analog and Digital Computers for
Industrial Application; Analog Components

See inside back cover for additional listings.

The charge for an Institutional Listing is $20.00 per issue or $60.00 for four consecutive issues. (The number of characters, including spaces, is limited to ninety for the company name and address, and ninety for the products and services.)

Applications for Institutional Listings and checks (made out to the Institute of Radio Engineers) should be sent to Technical Secretary, Institute of Radio Engineers, 1 East 79th St., New York 21, N. Y.